A UNIFYING FRAMEWORK FOR DISJUNCTIVE DATA CONSTRAINTS WITH APPLICATIONS TO REASONING UNDER UNCERTAINTY

Mathias Niepert

Submitted to the faculty of the University Graduate School
in partial fulfillment of the requirements
for the degree
Doctor of Philosophy
in the Department of Computer Science
Indiana University
June 2009

Accepted by the Graduate Faculty, Indiana University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Dirk Van Gucht, Ph.D.

Marc Gyssens, Dr.Sc.

Lawrence S. Moss, Ph.D.

David Leake, Ph.D.

Paul W. Purdom, Ph.D.

12 June 2009

Acknowledgements

There are many people who helped to make this thesis possible.

First, I want to thank my advisor Dirk Van Gucht. Without him, I would not have joined the PhD program at Indiana University and I would not have realized that research is exactly what I want and need to do in my professional life. If it weren't for his guidance, encouragement and mentoring, I would not be where I am today. Despite his many administrative duties that were given to him during the last four years, he has always reserved some time and energy for me, discussing research problems, and writing papers for conferences and journals. Marc Gyssens also contributed tremendously to my development as a scientist. He has been a great collaborator and contributed heavily to my dissertation research. His mathematical skills and insights have always impressed me. Doing research with Dirk and Marc has been a lot of fun and never something that I had to force myself to do. Over the years and during Marc's visits to Bloomington, we have become friends, and I hope that, in the future, we will be able to continue the good research as well as our friendship.

I am also deeply indebted to Colin Allen for his support and guidance. Even though I was not one of his own PhD students, he let me join the InPhO project, which has always been challenging and fun, and which has given me professional distraction when I needed it the most. Colin is one of the smartest, most loyal, and hardest working people I know. Doing research, travelling, and celebrating grants at his house was a lot of fun. I hope to be able to continue working and celebrating with him.

I also want to acknowledge the other members of my dissertation committee for their commitment and support. Larry Moss, Paul Purdom, and David Leake were always responsive and provided me with valuable feedback. They have helped me tremendously in improving the style and content of my dissertation.

I very much want to thank my parents Elke and Bernd, my brother Daniel, and the rest of the great Niepert family for their support. They let me go when I needed to explore the world and have always welcomed me warmly when I returned for visits. Their love and moral support gave me the strength and self-confidence that I needed to successfully finish my dissertation. I also want to thank my grandfather Fritz Niepert and my grandmother Gretel Otto for their inspiration, love and support. I wish they could have seen this document.

A big "thank you" also to the good friends I have found in Bloomington: Cameron Buckner, Lindita Camaj, Clemens Matuschek, Fabian Wettstein, Georgi Chunev, and many more. Their friendship has allowed me to live a surprising sociable life as a PhD student, and provided me with the companionship that kept me sane when work and stress seemed overwhelming.

Finally, I want to thank Grace Schwartz for everything she has done for me. There are not enough words to describe the support, love, and friendship she has given me over the last five years. I could not have made it without her and her wonderful family.

Abstract

Constraints on different manifestations of data are a central concept in numerous areas of computer science. Examples include mathematical logic, database systems (functional and multivalued dependencies), data mining (association rules), and reasoning under uncertainty (conditional independence statements). One is often interested in a process that derives all or most of the constraints that are entailed by a set of known ones, without the expense and error-proneness of repeatedly analyzing the data. This is what is generally known as the implication problem for data constraints. We present a theoretical framework for disjunctive data constraints and the associated implication problems based on the observation that many instances can be reduced to an implication problem for additive constraints on specific classes of real-valued functions. Furthermore, we provide inference systems and testable properties of classes of real-valued functions which imply the soundness and completeness of these systems. We also derive properties of classes of functions that imply the non-existence of finite, complete axiomatizations. The theoretical framework is applied to derive novel results in the areas of uncertain reasoning and graphical models.

Contents

List of Figures

CHAPTER 1

Introduction

1. Motivation

Many concepts in data mining, machine learning, and database research can be understood by examining, discovering, and reasoning about constraints on (or properties of) the data that is to be mined and modeled. In frequent pattern mining, for example, association rules often indicate constraints that hold in the data [**5**]. They often reveal unexpected and valuable knowledge about the data under investigation (such as patterns in log-files, transaction databases, and text documents). Another example are conditional independence statements which are used in reasoning under uncertainty to factorize joint probability distributions. Conditional independence statements play a fundamental role in machine learning approaches such as Markov and Bayesian networks [**31**]. Again, every conditional independence statement is a specific constraint on the probability distribution that one wants to model and query efficiently. In relational databases, functional dependencies are constraints that express a direct functional relationship between attributes of the database schema. For example, in a human resources database, a functional dependency can be used to model that every employee has a single unique employee identification number.

These examples illustrate the importance of data constraints – the more constraints we have knowledge about, the better we can understand and model the data. However, even though these data constraints play a crucial role in numerous areas, a clear theoretical framework that unifies these types of data constraints, syntactically and semantically, did not exist. If such a framework was to be developed, it had to be able to include a process which derives all or most of these constraints that are entailed by a set of known ones, without the expense and error-proneness of repeatedly analyzing the data. This is what we call the implication problem for data constraints. Analogous to implication problems in mathematical logic, the following two questions are pivotal: (1) does there exist a finite axiomatization of the implication problem using inference rules, and (2) what is the computational complexity of the implication problem? If the answer to the first question is positive, we would know that the implication problem is decidable and that there exists a finite set of inference rules that are sound (i.e., all the inferred constraints do in fact hold in the data) and complete (i.e., every entailed constraint can be inferred under the system). Finding answers to the second question is important because it would determine whether the use of inference rules has advantages over repeatedly analyzing the data. Furthermore, knowledge about the computational complexity of the implication problems opens the possibility of leveraging new and existing algorithms to efficiently compute more concise representations of the known constraints. While these questions are theoretically interesting and important, they also have significant practical ramifications. In data mining, for example, inference rules are successfully employed to increase the efficiency of association rule mining algorithms [**5**] and to compute compact representations of frequent patterns [**4**]. In the area of reasoning under uncertainty, inference systems for conditional independence statements have been successfully used to learn the structure of probabilistic graphical models [**13**]. The notion of conditional independence is at the heart of the foundations of probabilistic graphical models [**8, 31**] and other paradigms for representing and reasoning in artificial intelligence. In relational databases, inference rules are used to infer new dependencies from a given set of known ones, supporting optimal database design and physical data representations such as indexes. Functional and multivalued dependencies are also leveraged for more efficient query processing [**14**].

While studying this variety of domain applications, we noticed a syntactic commonality of the different types of constraints. In particular, we observed in many situations involving reasoning about data that the distribution of the data is restricted by constraints which can be specified syntactically as *disjunctive statements* of the form $X \to \{Y_1, \dots, Y_n\}$, with $X, Y_1, \dots, Y_n$ pairwise disjoint subsets of some finite set S. (We will call n the order of this disjunctive statement.) If $X \cup Y_1 \cup ... \cup Y_n = S$, we say that the disjunctive statements is *saturated*. In the remainder of this section, we will try to convince the reader of the widespread occurrence of *disjunctive statements* in applications by presenting a variety of more concrete examples from different application areas. We will show that disjunctive statements occur naturally in various areas such as data mining, database theory, reasoning under uncertainty, and propositional logic. In each of these areas, disjunctive statements are at the basis of methods and tools used by researchers and practitioners. Our theoretical framework, which we will introduce more formally in the following chapter, will unify these seemingly unrelated areas, and this broader scope will provide a new angle of attack on existing open problems and future investigations. Indeed, we will demonstrate how the framework allowed us to solve some open problems in these areas. Furthermore, this unifying framework will enable cross-fertilization between these different areas: a concept developed in one area can be transformed into a corresponding concept in another area. Despite this variety of application areas, we will be able to show that, in all these instances, disjunctive statements can be interpreted as difference equations [**16**] of real-valued functions set to zero, a notion which we will refer to as *additive constraints* on real-valued functions.

Let us start by presenting some examples in which disjunctive statements occur syntactically. In the context of **frequent pattern mining**, let S be a set of *items*. For example, in a grocery store application, this set would represent items that can be bought by a customer. By definition, a *basket database* $\mathcal{B}$ is a set of subsets of S. Each subset (basket) represents a set of items bought by a shopper and $\mathcal{B}$ is the set of all shoppers' baskets. One is often interested in finding associations and correlations between sets of items. For instance, knowing that a customer who buys bread often also buys milk might be useful information for the vendor. We say that a basket database $\mathcal{B}$ satisfies the *generalized disjunctive rule* $X \to Y_1 \ \vee \ldots \vee \ Y_n$, with $X, Y_1, \ldots, Y_n$ pairwise disjoint subsets of S, if for each basket B in $\mathcal{B}$ which contains all items in X, there is some i, $1 \leq i \leq n$, such that B also contains all items in Y_i. In our example, this means that, whenever a customer bought the set of items X, then he also bought at least one of the sets of items $Y_1, \ldots, Y_n$. The generalized disjunctive rule above can be re-interpreted as a disjunctive statement of order n. *Association rules* and *disjunctive rules* are generalized disjunctive rules of order 1 and order 2 [**5**],

$$\mathbf{r} := \begin{array}{|c|c|c|c|} \hline \mathbf{a} & \mathbf{b} & \mathbf{c} & \mathbf{d} \\ \hline 1 & 1 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 \\ \hline 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ \hline \end{array}$$

FIGURE 1.1. A basket database $\mathcal{B}$ over the set of items $S = \{a, b, c, d\}$. $\mathcal{B}$ satisfies the association rule constraint (AR) $\{c\} \rightarrow \{a\}$ and the disjunctive rule constraint $\{a\} \rightarrow \{b\} \lor \{c\}$.

respectively. For example, the basket database in Figure 1.1 satisfies the association rule constraint $\{c\} \rightarrow \{a\}$ and the disjunctive rule constraint $\{a\} \rightarrow \{b\} \lor \{c\}$. Association rules and disjunctive association rules play an important role in frequent itemset mining. The objective here is to mine for itemsets that, relative to some threshold value, are frequently contained in the baskets of a basket database. These rules are particularly useful to prune the search space of the frequent itemsets (i.e., sets of items that are frequently bought together), leading to a succinct representation of the set of all frequent itemsets [**4**]. The rationale behind pruning the frequent itemsets search space is that, if association rules or disjunctive rules are known to exist, then the frequency of certain itemsets can be inferred from the known frequency of others, and this without having to count their frequency explicitly in the baskets database. Indeed, if $\mathcal{B}$ satisfies the association rule $X \rightarrow Y$, then, from knowing how many baskets contain X, we can infer the number of baskets that contain $X \cup Y$.

In the context of **relational databases**, all data is viewed as being stored in tables which are called relations. The rows of these relations are referred to as tuples and they summarize some object or relationship in the real world [**25**]. Each relation in the database is specified by a schema, which is a set of attributes that correspond to the columns of the relation. A relation instance is a particular instantiation of such a relation schema,

$\mathbf{r} :=$

a	**b**	**c**	**d**
a1	b1	c1	d1
a1	b1	c1	d2
a1	b2	c1	d1
a1	b2	c1	d2
a2	b1	c2	d1

FIGURE 1.2. A relation instance $\mathbf{r}$ over the set of attributes $S = \{a, b, c, d\}$. The instance $\mathbf{r}$ satisfies the functional dependency $\{a\} \to \{c\}$ which can be re-written as the disjunctive statement $\{a\} \to \{\{c\}\}$. The instance also satisfies the MVD $\{a\} \to \{b\} \mid \{c, d\}$ which can be re-written as the disjunctive statement $\{a\} \to \{\{b\}, \{c, d\}\}$.

that is, a set of tuples over that schema. We recall that a relation instance $\mathbf{r}$ satisfies the functional dependency $X \to Y$ if, for any tuples t_1 and t_2 in $\mathbf{r}$, $t_1[X] = t_2[X]$ implies $t_1[Y] = t_2[Y]$.[1] A relation instance $\mathbf{r}$ satisfies the multivalued dependency $X \twoheadrightarrow Y|Z$, with $X \cup Y \cup Z = S$, if $\mathbf{r}$ can be losslessly decomposed into its projections on $X \cup Y$ and $X \cup Z$, that is, if $\mathbf{r} = \prod_{X \cup Y}(\mathbf{r}) \bowtie \prod_{X \cup Z}(\mathbf{r})$, where $\prod_X(\mathbf{r})$ is the projection of $\mathbf{r}$ on X and $\bowtie$ is the natural join operator.[2] This decomposition removes redundancy and prevents update anomalies, thus leading to a better database design. In the context of relation instances over a schema S, *functional* and *multivalued dependencies* can be interpreted as disjunctive statements of order 1 and order 2, respectively. Indeed, the FD $X \to Y$ can be re-written as the disjunctive statement $X \to Y$, and the MVD $X \to Y \mid Z$ can be re-written as the disjunctive statement $X \to \{Y, Z\}$. Figure 1.2 illustrates the concepts above.

Let S be a set of random variables in the context of **probability theory** and reasoning under uncertainty. A *probability measure* P satisfies a *conditional independence statement*

---

[1] $t_1[Y]$ represents the content of tuple t_1 restricted to the attribute set Y.

[2] Note that the functional dependency $X \to Y$ logically implies the multivalued dependency $X \twoheadrightarrow Y|Z$, for any Z with $X \cup Y \cup Z = S$.

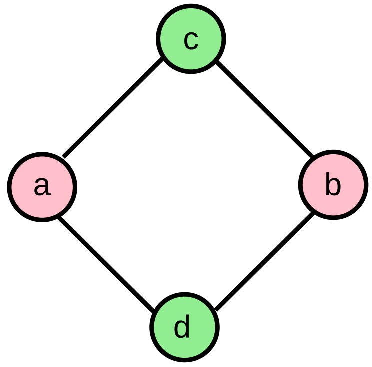

FIGURE 1.3. A Markov network over a set of statistical variables $S = \{a, b, c, d\}$. Different separation criteria in graphical models "encode" conditional independence statements. For example, the Markov network shown above encodes the CI statements $I(a, b|\{c, d\})$ and $I(c, d|\{a, b\})$.

$I(Y, Z|X)$, with X, Y, and Z pairwise disjoint subsets of S, if, for every assignment $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ to the variables in X, Y, and Z, respectively, $P(\mathbf{x})P(\mathbf{x}, \mathbf{y}, \mathbf{z}) = P(\mathbf{x}, \mathbf{y})P(\mathbf{x}, \mathbf{z})$ [**15**]. Conditional independence (CI) is a central concept in reasoning about uncertainty [**17**], because it allows for more compact representations of probability distributions. The conditional independence statement $I(Y, Z|X)$ can be rewritten as the disjunctive statement $X \to \{Y, Z\}$ of order 2. Figure 1.3 depicts a Markov network representing a set of CI statements over four random variables. We will discuss conditional independence statements and the applications of the presented theory in the area of reasoning under uncertainty in Chapter 5.

We next consider **propositional logic**, which is extensively used in numerous applications. Let S be a set of *propositional variables*. A *truth assignment* $\mathbf{w}$ over S satisfies an *implication formula* $X \to Y$, with X and Y disjoint subsets of S, if $\mathbf{w}$ satisfies the propositional formula $\bigwedge_{x \in X} x \to \bigvee_{y \in Y} y$. In the syntax of disjunctive statements, the implication formula $X \to Y$ takes the form $X \to \{\{y\} \mid y \in Y\}$. Its order is $|Y|$. Notice that each propositional formula can be rewritten into a conjunction of implication formulae. For example, the formula in conjunctive normal form $(p \vee q \vee \neg r \vee \neg s) \wedge (p \vee r) \wedge (\neg q \vee \neg s)$ is equivalent to $(r \wedge s \to p \vee q) \wedge (\mathbf{true} \to p \vee r) \wedge (q \wedge s \to \mathbf{false})$. The conjuncts correspond to the implication formulae $\{r, s\} \to \{p, q\}$, $\emptyset \to \{p, r\}$, and $\{q, s\} \to \emptyset$, respectively. Finally, these implication formulae correspond to the disjunctive statements $\{r, s\} \to \{\{p\}, \{q\}\}$, $\emptyset \to \{\{p\}, \{r\}\}$, and $\{q, s\} \to \emptyset$

Our final example concerns **cooperative game theory**. The main objective of cooperative game theory is to provide a formal framework for reasoning about multi-player games in which players can form coalitions for joint cooperations [**23, 1**]. Let S be a non-empty set of players. An *interaction statement* over S has the form $X \to \{i, j\}$, with X a set of players, and i and j two other players in S. A *worth function* w assigns to each subset X of S a worth value $w(X)$ representing the combined worth of the coalition X in the game. It is generally assumed that $w(\emptyset) = 0$. We say that w satisfies $X \to \{i, j\}$ if player i and j act without interference when joining the coalition X in the game. This is formally defined by requiring that w satisfies the additive constraint $w(X \cup \{i, j\}) - w(X \cup \{i\}) - w(X \cup \{j\}) + w(X) = 0$. In the syntax of disjunctive statements, the interference statement $X \to \{i, j\}$ takes the form $X \to \{\{i\}, \{j\}\}$. Its order is 2. Various classes of cooperative games have been considered on the basis of certain properties of their worth functions. For instance, a worth function w is *convex* (or supermodular) if $w(X \cup Y) + w(X \cap Y) \geq w(X) + w(Y)$, and *concave* (or submodular) if $w(X \cup Y) + w(X \cap Y) \leq w(X) + w(Y)$, in both cases for each $X, Y \subseteq S$.

2. Disjunctive Statements as Constraints on Real-Valued Functions

In the previous section, we presented a wide variety of examples in which data are restricted by constraints which can be formalized syntactically as disjunctive statements. Despite this syntactic uniformity, the types of data on which these constraints operate are extremely diverse and, therefore, also their semantics. However, semantic uniformity can be achieved by characterizing the various constraints in terms of additive constraints on real-valued functions associated with the data, as we will demonstrate next.

We first return to **frequent pattern mining**. Given a set of items S and a basket database $\mathcal{B}$, the *support function* $\mathtt{support}_{\mathcal{B}}$ is the function that associates with each subset X of S the number of baskets in $\mathcal{B}$ that include X, i.e., $\mathtt{support}_{\mathcal{B}}(X) = |\{B \in \mathcal{B} \mid X \subseteq B\}|$. It can now be verified that $\mathcal{B}$ satisfies the generalized disjunctive rule $X \rightarrow Y_1 \ \vee \ldots \ \vee \ Y_n$ if $\sum_{\mathcal{Z} \subseteq \mathcal{Y}} (-1)^{|\mathcal{Z}|} \mathtt{support}_{\mathcal{B}}(X \cup \bigcup_{Z \in \mathcal{Z}} Z) = 0$, where $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$ [**38**]. In particular, $\mathcal{B}$ satisfies the association rule constraint $X \rightarrow Y$ if $\mathtt{support}_{\mathcal{B}}(X) - \mathtt{support}_{\mathcal{B}}(X \cup Y) = 0$ [**40, 38**]. Similarly, $\mathcal{B}$ satisfies the disjunctive rule constraint $X \rightarrow Y \ \vee \ Z$ if and only if $\mathtt{support}_{\mathcal{B}}(X) - \mathtt{support}_{\mathcal{B}}(X \cup Y) - \mathtt{support}_{\mathcal{B}}(X \cup Z) + \mathtt{support}_{\mathcal{B}}(X \cup Y \cup Z) = 0$ [**40, 38**].

We next come back to **relational databases**. Given a relation schema S and a relation instance r over S, we define a real-valued function H_r as follows. Let P be the uniform probability distribution over the tuples of r, i.e., for each tuple t in r, $P(t) = 1/|r|$. The *Shannon entropy* H_s is defined by

$$H_r(X) = - \sum_{x \in \pi_X(s)} P^X(x) \log(P^X(x)),$$

where $P^X(x) = \sum_{t \in s \ \& \ t[X]=x} P(t)$ is the *marginal probability measure* induced by P on X. It can be shown that r satisfies the functional dependency $X \rightarrow Y$ if and only if $H_r(X) - H_r(X \cup Y) = 0$ [**26, 21, 7**]. Similarly, it can be show that r satisfies the multivalued dependency $X \twoheadrightarrow Y|Z$ if and only if $H_r(X) - H_r(X \cup Y) - H_r(X \cup Z) + H_r(X \cup Y \cup Z) = 0$ [**26, 21, 7**].

Let us now return to **reasoning under uncertainty**. Given a set S of random variables and a probability measure P over S, we associate a real-valued function with P as follows. Let H_P be the relative entropy (Kullback-Leibler divergence) [**20**]. The *multi-information function* $M_P : 2^S \rightarrow [0, \infty)$ is defined by $M_P(\emptyset) = 0$ and

$$M_P(X) = H_P(P^X | \prod_{x \in X} P^{\{x\}}),$$

for each non-empty subset X of S. As in the previous example, P^X and $P^{\{x\}}$ represent marginal probability measures. It can now be shown that P satisfies the conditional independence statement $I(Y, Z|X)$ if and only if $M_P(X) - M_P(X \cup Y) - M_P(X \cup Z) + M_P(X \cup Y \cup Z) = 0$ [**46**].

Let us again consider **propositional logic**. Given a set S of propositional variables and a truth-assignment $\mathbf{w}$ over S, we define the real-valued function $W_{\mathbf{w}}$ by $W_{\mathbf{w}}(X) = 1$ if each variable in X evaluates to **true** under $\mathbf{w}$, and $W_{\mathbf{w}}(X) = 0$ otherwise. It can now be shown that $\mathbf{w}$ satisfies the implication formula $X \to Y$ if and only if $\sum_{\mathcal{Z} \subseteq \mathcal{Y}} (-1)^{|\mathcal{Z}|} W_{\mathbf{w}}(X \cup \bigcup_{Z \in \mathcal{Z}} Z) = 0$, where $\mathcal{Y} = \{\{y\} \mid y \in Y\}$.

Finally, notice that, in **cooperative game theory**, interaction statement constraints on worth functions are already defined as additive constraints.

3. Implication Problems and Inference Systems

The consideration of constraints in applications such as the ones we have previously discussed leads naturally to the *implication problems* for these constraints. For example, it is known that if a relation instance satisfies the functional dependencies $X \to Y$ and $Y \to Z$, it also satisfies the functional dependency $X \to Z$. More generally, the implication problem is, given some set S over which the constraints are defined, a set $\mathcal{C}$ of disjunctive statements over S, and a single disjunctive statement c over S, to decide whether $\mathcal{C} \models c$, i.e., whether each data set satisfying all disjunctive statements in $\mathcal{C}$ also satisfies c.

Deciding instances of the implication problem is at the heart of the applications identified in the examples. A classic way to decide or to facilitate decision of these implication problems is to design and apply *inference systems* that are sound, or complete, or, preferably, both. When we revisit the previous examples, we observe that the inference systems introduced for these in the literature have a significant amount of syntactic commonality. For instance, in Figure 1.4, we exhibit a sound and complete inference system for functional dependencies and association rules [**40**]; in Figure 1.5, we exhibit a sound and complete inference system for multivalued dependencies [**2**]; in Figure 1.6, we exhibit a sound, but not complete, inference system for conditional independence statements [**33, 46**]; and in Figure 1.7, we exhibit a sound and complete inference system for implication formulae.

Clearly, the semantic definitions of the constraints involved do not share a strong resemblance. Nevertheless, it is possible to reformulate them as additive constraints on specific classes of real-valued functions associated with the data. This observation is a key motivation to formally introduce a generalized unifying framework for disjunctive statements. By doing so, we will also achieve semantic uniformity at the level of the implication problems.

4. Outline

In Chapter 2, we formally define real-valued functions, differentials, semi-lattice decompositions, and the Möbius transform, all of which are notions employed to develop the theoretical framework for disjunctive statements. We show how differentials of a real-valued function correspond to sums of elements of its densities specified by the Möbius transform and semi-lattice decompositions. We introduce the important class of Choquet capacities, a generalization of supermodular and submodular real-valued functions which occur in many of the applications covered in this dissertation. We show how disjunctive statements can be interpreted as additive constraints on real-valued functions. Finally, we introduce the associated implication problem.

In Chapter 3, we present inference systems that are sound and/or complete for several well-known implication problems. These inference systems serve as *templates* for specialized inference systems in the context of implication problems with bounds on the order of the disjunctive statements. We prove these inference systems to be sound and complete with respect to semi-lattice inclusion, that is, we show that if the semi-lattice of a disjunctive statement is a subset of the union of semi-lattices of a set of CI statements, then the single CI statement can be derived from the set of CI statements. We harness this characterization to derive testable properties that imply the soundness and/or completeness of the two inference systems for instances of the implication problem.

Reflexivity

$$\frac{}{X \supseteq Y : X \to Y}$$

Transitivity

$$\frac{X \to Y \qquad Y \to Z}{X \to Z}$$

Augmentation

$$\frac{X \to Y}{X \cup W \to Y}$$

Union

$$\frac{X \to Y \qquad X \to Z}{X \to Y \cup Z}$$

FIGURE 1.4. Sound and complete inference system for functional dependencies and association rules.

Triviality

$$\frac{}{X \rightarrow \emptyset \mid \overline{X}}$$

Transitivity

$$\frac{X \rightarrow Y \mid W \qquad Y \rightarrow Z \mid V}{X \rightarrow Z - Y \mid W \cup (Y \cap V)}$$

Augmentation

$$\frac{X \rightarrow Y \cup W \mid Z}{X \cup W \rightarrow Y \mid Z}$$

Symmetry

$$\frac{X \rightarrow Y \mid Z}{X \rightarrow Z \mid Y}$$

FIGURE 1.5. Sound and complete inference system for multivalued dependencies.

Triviality

$$\frac{}{I(\emptyset, \overline{X} | X)}$$

Contraction

$$\frac{I(Y, Z | X) \quad I(Y, W | X \cup Z)}{I(Y, Z \cup W | X)}$$

Weak Augmentation

$$\frac{I(Y, Z \cup W | X)}{I(Y, Z | X \cup W)}$$

Decomposition

$$\frac{I(Y, Z \cup W | X)}{I(Y, Z | X)}$$

FIGURE 1.6. Sound, but not complete, inference system for conditional independence statements.

Augmentation

$$\frac{X \to Y}{X \cup W \to Y}$$

Strong Resolution

$$\frac{\begin{array}{c} X \to Y \cup Z \\ \forall z \in Z : X \cup \{z\} \to Y \cup V \end{array}}{X \to Y \cup V}$$

Composition

$$\frac{\begin{array}{c} X \to Y \\ X \to Z \end{array}}{X \to Y \cup Z}$$

Decomposition

$$\frac{X \to Y \cup Z}{X \to Y}$$

FIGURE 1.7. Sound and complete inference system for implication formulae.

In Chapter 4, we specify properties of real-valued functions that imply the non-existence of a finite axiomatization for the implication problem for disjunctive statements interpreted as additive constraints on specific classes of real-valued functions. These properties, which hold for many classes of real-valued functions, allow us to solve some open problems in several areas of computer science. The theory builds on, and generalizes, previous ideas of non-existence proofs in the areas of embedded multivalued dependencies [**35**] and conditional independence in reasoning under uncertainty [**44**]. We show the theoretical framework to be applicable to a variety of instances of the implication problem for disjunctive statements.

In Chapter 5, we employ the framework within the area of reasoning under uncertainty. Using the previously developed theory, we show that an extension of the semi-graphoid axioms is (1) sound and complete for saturated CI statements, (2) complete for general CI statements, and (3) sound and complete for stable CI statements (de Waal and van der Gaag [**9**]), all relative to the class of discrete probability measures. By interpreting conditional independence statements as disjunctive statements and applying the theoretical framework, we first prove that the inference system is sound and complete relative to certain inclusion relationships on the involved semi-lattices. To make the connection between the theoretical framework for disjunctive statements and the conditional independence implication problem, we discuss the concept of *multi-information functions* induced by probability measures (Studený [**46**]). This class of real-valued functions allows us to link the implication problem for additive constraints on real-valued functions to the probabilistic CI implication problem. We introduce an approximate logical inference algorithm that combines a powerful falsification algorithm and a novel validation algorithm which represents implication problems as instances of linear programming problems. We show experimentally that the falsification and validation criteria, some of which can be tested in polynomial time, work very effectively. We will relate the experimental results to those obtained for the existing *racing algorithm* introduced by Bouckaert and Studený [**3**].

Finally, in Chapter 6, we harness the theoretical framework to further investigate the logical and algorithmic properties of stable conditional independence structures, an important concept for representing and reasoning about conditional independence information.

We show that stable CI structures are a powerful generalization of Markov networks. Furthermore, the theory establishes a direct connection between sets of stable CI statements and propositional formulae in conjunctive normal form. We leverage this connection to prove that the implication problem for stable conditional independence is coNP-complete. We derive a linear time reduction to the Boolean satisfiability problem (SAT), and verify empirically that existing SAT solvers can be used to efficiently decide the implication problem and to compute concise, non-redundant representations of stable CI structures, even for instances involving hundreds of variables.

5. Related Work

Some of the concepts we will use throughout the thesis are based on previous work by Bassem Sayrafi, Dirk Van Gucht, Marc Gyssens, and Paul Purdom [**36, 37, 39, 38, 40**]. While the previous work has specifically focused on applications in data mining and initiated the development of the lattice-theoretic framework, this work focuses on disjunctive statements interpreted as additive constraints with bounded order, making the theory applicable to existing concepts in various areas of computer science. Bassem Sayrafi's work focused primarily on the class of frequency functions over finite databases [**38**]. The derived inference rules were used in frequent itemset mining algorithms as heuristics to prune the search space [**41**]. Among the first to discover a relationship between functional dependencies in databases and the mathematical notion of a lattice were János Demetrovics, Leonid Libkin, and Ilya B. Muchnik [**11**]. Our work was additionally influenced by Mehmet Dalkilic and Edward Robertson's theoretical investigation of information dependencies in databases [**7**] and Francesco Malvestuto's work on information content of a database [**26**]. A rigorous investigation and classification of properties of real-valued functions based on their differentials, was first undertaken in Gustave Choquet's seminal monograph [**6**].

In more recent work [**30, 29, 28**], we extended and applied the theory in the context of probabilistic systems, analyzing the conditional independence implication problem. Of course, a vast amount of related work existed in this area. The work with the greatest impact on our research was Milan Studený's monograph on structural representations of conditional independence structures [**46**], Dan Geiger and Judea Pearl's paper about the algorithmic and logical properties of conditional independence [**15**], and Peter de Waal and Linda van der Gaag's paper introducing the notion of stable independence [**9**]. Conditional independence structures are algorithmically challenging since their combinatorics are of super-exponential magnitude, rendering some of the associated computational problems intractable. For example, algorithms existed only for instances of the implication problem for at most six variables [**3**]. Using our theory, we were able to increase the number of variables of computable instances to more than 20 by improving both the effectiveness and efficiency of the algorithms [**28**]. We also succeeded in gaining deeper insights into some of the underlying theoretical problems and, as a consequence, were able to settle some existing open problems.

6. Publications

Some of the content of this thesis has appeared in previous publications.

- Mathias Niepert. **Logical Inference Algorithms and Matrix Representations for Probabilistic Conditional Independence.** In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence* (UAI), Montreal, Canada, 2009 (to appear).
- Mathias Niepert and Dirk Van Gucht. **Logical Properties of Stable Conditional Independence.** In *Proceedings of the 4th European Workshop on Probabilistic Graphical Models*, Hirtshals, Denmark, pages 225–232, 2008. (The paper was selected for a special issue of the *International Journal of Approximate Reasoning*)
- Mathias Niepert, Dirk Van Gucht, and Marc Gyssens. **On the Conditional Independence Implication Problem: A Lattice-Theoretic Approach.** In *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*, Helsinki, Finland, pages 435–443, AUAI Press, 2008. (**Best student paper runner-up award**; extended version will be submitted to the journal *Artificial Intelligence* following an invitation).

CHAPTER 2

A Unifying Framework for Disjunctive Statements

In the previous chapter, we have observed that disjunctive statements occur both syntactically and semantically in various disciplines such as computer science, artificial intelligence, and game theory. We will now formalize this observation by introducing a general theoretical framework for disjunctive statements. We will first formally define disjunctive statements. Since we will interpret disjunctive statements as *additive constraints* on real-valued functions, we also introduce the notion of an additive constraint as well as several other important concepts related to real-valued functions, such as the concept of the Möbius transform, differentials on real-valued functions, and Choquet capacities [**6**]. The goal of this chapter is to provide a clear theoretical framework that captures a wide variety of implication problems both syntactically and semantically.

1. Real-valued Functions, Differentials, and Densities

Definition 2.1 (Real-valued function). Let S be a finite set. A *real-valued function* $F : 2^S \to \mathbf{R}$ *over* S associates a real number to each subset of S.

Definition 2.2 (Differential). Let S be a finite set, let $F : 2^S \to \mathbf{R}$ be a real-valued function, and let $\mathcal{Y}$ be a set of subsets of S. The $\mathcal{Y}$-*differential* of F is the function $\Delta^{\mathcal{Y}}F : 2^S \to \mathbf{R}$, and is defined recursively by

$$\begin{aligned}\Delta^{\emptyset}F(X) &= F(X);\text{ and}\\ \Delta^{\mathcal{Y}\cup\{Y\}}F(X) &= \Delta^{\mathcal{Y}}F(X) - \Delta^{\mathcal{Y}}F(X \cup Y),\end{aligned}$$

for each $X \subseteq S$.

Note the analogy with the definition of derivatives of real functions. Here, $|\mathcal{Y}|$ corresponds to the order of differentiation.

EXAMPLE 2.3. According to Definition 2.2, we have, for a finite set S, a function $F : 2^S \to \mathbf{R}$, and $X, Y, Z \subseteq S$, that

$$\begin{aligned}
\Delta^{\emptyset} F(X) &= F(X); \\
\Delta^{\{Y\}} F(X) &= F(X) - F(X \cup Y); \text{ and} \\
\Delta^{\{Y,Z\}} F(X) &= F(X) - F(X \cup Y) - F(X \cup Z) + F(X \cup Y \cup Z).
\end{aligned}$$

Notice that the last expression is symmetric in Y and Z. For a formal proof of the well-definedness of differentials, we refer to Sayrafi and Van Gucht [**38**].

We will now define the notion of a *density* of a real-valued function. Alternative terms used in the literature are Möbius inversion or Möbius transform.

DEFINITION 2.4 (Density). Let S be a finite set and let F be a real-valued function over S. The *density* of F is the real-valued function over S defined by

$$\Delta F(X) = \sum_{X \subseteq U \subseteq S} (-1)^{|U|-|X|} F(U),$$

for each $X \subseteq S$.

EXAMPLE 2.5. Let $S = \{a, b, c\}$ and let F be a real-valued function over S. Then,

$$\begin{aligned}
\Delta F(\{a,b,c\}) &= F(\{a,b,c\}); \\
\Delta F(\{a,b\}) &= F(\{a,b\}) - F(\{a,b,c\}); \text{ and} \\
\Delta F(\{a\}) &= F(\{a\}) - F(\{a,b\}) - F(\{a,c\}) + F(\{a,b,c\}).
\end{aligned}$$

2. Semi-lattice Decompositions

We will later derive and analyse properties of classes of real-valued function and relate these properties to soundness and completeness properties of inference systems. An important concept needed for this purpose are so-called semi-lattices associated with disjunctive statements. Let us begin with the definition of a lattice spanned by two sets. For a finite set S and $X, Y \subseteq S$, the lattice $[X, Y]$ is defined as $\{U \mid X \subseteq U \subseteq Y\}$. Note that, by definition, $[X, Y] = \emptyset$ if $X \nsubseteq Y$. We can now define the notion of a semi-lattice decomposition of a disjunctive statement.

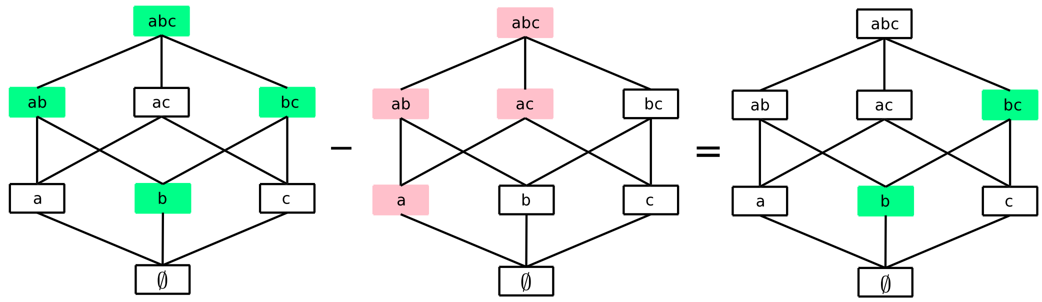

FIGURE 2.1. Let $S = \{a, b, c\}$ be a finite set. Then the semi-lattice decomposition of $\{b\}$ relative to $\{\{a\}\}$ is equal to $\{\{b\}, \{b, c\}\}$, or, equivalently, $\mathcal{L}(\{b\}, \{\{a\}\}) = \{\{b\}, \{b, c\}\}$.

DEFINITION 2.6 (Semi-Lattice decomposition). Let S be a finite set, let $X \subseteq S$, and let $\mathcal{Y}$ be a set of subsets of S. The *semi-lattice decomposition* of X *relative* to $\mathcal{Y}$ are the supersets of X which are not supersets of any element in $\mathcal{Y}$. More formally, the semi-lattice decomposition of X relative to $\mathcal{Y}$ is defined by

$$\mathcal{L}(X, \mathcal{Y}) = [X, S] - \left(\bigcup_{Y \in \mathcal{Y}} [Y, S] \right).$$

EXAMPLE 2.7. Let $S = \{a, b, c\}$ be a finite set. Then $\mathcal{L}(\{b\}, \{\{a\}\}) = [\{b\}, \{a, b, c\}] - [\{a\}, \{a, b, c\}] = \{\{b\}, \{b, c\}\}$. Figure 2.1 illustrates how the semi-lattice is constructed using Hasse diagrams.

EXAMPLE 2.8. Let $S = \{a, b, c, d\}$ be a finite set. Then,

$$\begin{aligned}
\mathcal{L}(\{a\}, \{\{b, c\}, \{d\}\}) &= [\{a\}, \{a, b, c, d\}] - ([\{b, c\}, \{a, b, c, d\}] \cup [\{d\}, \{a, b, c, d\}]) \\
&= \{\{a\}, \{a, b\}, \{a, c\}\}; \\
\mathcal{L}(\emptyset, \{\{a, b\}, \{c, d\}\}) &= [\emptyset, \{a, b, c, d\}] - ([\{a, b\}, \{a, b, c, d\}] \cup [\{c, d\}, \{a, b, c, d\}]) \\
&= \{\emptyset, \{a\}, \{b\}, \{c\}, \{d\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}\}; \text{ and} \\
\mathcal{L}(\{a\}, \{\{b, c\}, \emptyset\}) &= [\{a\}, \{a, b, c, d\}] - ([\{b, c\}, \{a, b, c, d\}] \cup [\emptyset, \{a, b, c, d\}]) \\
&= \emptyset.
\end{aligned}$$

For a disjunctive statement $c = X \to \mathcal{Y}$, we will sometimes write $\mathcal{L}(c)$ instead of $\mathcal{L}(X, \mathcal{Y})$. Similarly, for a set of disjunctive statements $\mathcal{C}$, we will sometimes write $\mathcal{L}(\mathcal{C})$

instead of $\bigcup_{c' \in \mathcal{C}} \mathcal{L}(c')$. The semi-lattice decomposition of a disjunctive statement can be used to express differentials induced by the disjunctive statements in terms of the densities of the real-valued functions [**38**].

PROPOSITION 2.9. *Let S be a finite set, let F be a real-valued function over S, and let $X \subseteq S$. Then,*

$$\Delta^{\mathcal{Y}} F(X) = \sum_{U \in \mathcal{L}(X,\mathcal{Y})} \Delta F(U).$$

EXAMPLE 2.10. Let $S = \{a, b, c, d\}$ be a finite set. Then,

$$\begin{aligned}
\Delta^{\{\{b,c\},\{d\}\}} F(\{a\}) &= F(\{a\}) - F(\{a,b,c\}) - F(\{a,d\}) + F(\{a,b,c,d\}) \\
&= \Delta F(\{a\}) + \Delta F(\{a,b\}) + \Delta F(\{a,c\}); \\
\Delta^{\{\{a,b\},\{c,d\}\}} F(\emptyset) &= F(\emptyset) - F(\{a,b\}) - F(\{c,d\}) + F(\{a,b,c,d\}) \\
&= \Delta F(\emptyset) + \Delta F(\{a,c\}) + \Delta F(\{a,d\}) + \Delta F(\{b,c\}) + \Delta F(\{b,d\}); \\
\Delta^{\{\{b,c\},\emptyset\}} F(\{a\}) &= F(\{a\}) - F(\{a,b,c\}) - F(\{a\}) + F(\{a,b,c\}) \\
&= 0.
\end{aligned}$$

We now introduce a generalization of the supermodularity and submodularity properties, respectively, of a class of real-valued functions.

DEFINITION 2.11 (Choquet capacities [**6**]). Let S be a finite set, let F be a real-valued function over S, and let k be a natural number, $k \geq 1$.

- The function F is a *positive k-alternating capacity* if, for each subset X of S and for each non-empty set $\mathcal{Y}$ of k subsets of S, $\Delta^{\mathcal{Y}} F(X) \geq 0$.
- The function F is a *negative k-alternating capacity* if, for each subset X of S and for each non-empty set $\mathcal{Y}$ of k subsets of S, $\Delta^{\mathcal{Y}} F(X) \leq 0$.

Please note that the notion of a positive (negative) 1-alternating capacity is equivalent to the notions of decreasing (increasing) real-valued function. Analogously, the notion of a positive (negative) 2-alternating capacity is equivalent to the notion of supermodular (submodular) real-valued function. Choquet capacities occur in areas such as game theory, fuzzy measures, belief and plausibility functions, and possibility measures. We will show in later chapters that classes of functions that are Choquet capacities have some interesting properties with respect to inference systems for the implication problem for disjunctive statements. For an introduction to, and discussion of, Choquet capacities in the area of reasoning about uncertainty, we refer the reader to Halpern [**17**].

3. Disjunctive Statements as Additive Constraints

DEFINITION 2.12 (Disjunctive statement). Let S be a finite set. A *disjunctive statement* over S is an expression of the form $X \to \mathcal{Y}$ with $\{X\} \cup \mathcal{Y}$ a set of pairwise disjoint subsets of S.[1] The size of $\mathcal{Y}$, $|\mathcal{Y}|$, is called the *order* of $X \to \mathcal{Y}$. If $X \cup \bigcup \mathcal{Y} = S$, the statement is called *saturated*.

EXAMPLE 2.13. Let $S = \{A, B, C, D\}$. The following are disjunctive statements of order 0, 1, 2, and 3, respectively:

$\{A, B, C\} \to \emptyset$;
$\{C\} \to \{\{A, B\}\}$;
$\{A, B\} \to \{\{C\}, \{D\}\}$; and
$\emptyset \to \{\{A, B\}, \{C\}, \{D\}\}$.

Disjunctive statements can be interpreted as additive constraints on real-valued functions.

DEFINITION 2.14 (Disjunctive statements as additive constraints). Let S be a finite set, let $X \subseteq S$ and $\mathcal{Y} \subseteq 2^S$ such that $\{X\} \cup \mathcal{Y}$ consists of pairwise disjoint sets, and let F be a real-valued function over S. Then F satisfies the *disjunctive statement* $X \to \mathcal{Y}$ if and only if $\Delta^{\mathcal{Y}} F(X) = 0$.

[1]In some applications, overlap between X and the members of $\mathcal{Y}$ is possible. All results still hold in this case, provided that the triviality rule in inference systems $\mathcal{K}$ (Figure 3.1) and $\mathcal{G}$ is replaced by $X \supseteq Y : X \to \mathcal{Y} \cup \{Y\}$. The proofs require only minor modifications.

We can now define the bounded logical implication problem for disjunctive statements with respect to a class of real-valued functions.

For a finite set S, a set of disjunctive statements $\mathcal{C}$ over S, a disjunctive statement c over S, and a class of real-valued functions $\mathcal{F}$ over S, we say that $\mathcal{C}$ *logically implies* c relative to $\mathcal{F}$, and write $\mathcal{C} \models_{\mathcal{F}} c$, if every function $F \in \mathcal{F}$ that satisfies all disjunctive statements in $\mathcal{C}$ also satisfies c.

By the *implication problem*, we mean the problem of deciding logical implication with S as a parameter of the problem. This means that, for each finite set S, an appropriate set of real-valued functions of the form $F : 2^S \to \mathbf{R}$ is given. This leads to the following definition.

Definition 2.15 (Class of functions). A class of real-valued functions $\mathcal{F}$ is a mapping that associates to each finite set S a set of real-valued functions over S. Whenever $\mathcal{F}$ is a class of real-valued functions, and S is a finite set, $\mathcal{F}_S$ denotes the set of real-valued functions over S associated to S by $\mathcal{F}$.

Definition 2.15 allows us to speak about the implication problem relative to a class of real-valued functions. In many cases, bounds are imposed on the order of the disjunctive statements under consideration. For natural numbers ℓ and u, $1 \leq \ell \leq u$, we mean by the $[\ell, u]$-*bounded implication problem* (or *bounded implication problem* if the bounds are implicit) the special case of the implication problem where only disjunctive statements of order at least ℓ and at most u are considered. It follows from the examples in the introduction that the bounded implication problem for disjunctive statements is a unifying framework for a wide variety of seemingly unrelated implication problems that have been considered in the literature.

The theoretical framework presented here unifies important implication problems occurring in computer science. We have shown that many concepts such as functional dependencies (FDs), association rules (ARs), multivalued dependencies (MVDs), and conditional independence statements (CIs) can be rewritten as disjunctive statements having a particular order. For example, in the context of conditional independence statements, the

Application-specific semantics (FDs, ARs, ...):	$\Leftrightarrow$	Corresponding classes of real-valued functions $(\mathcal{F}_{FDs}, \mathcal{F}_{ARs}, ...)$ and implicit bounds on the order; here: $[1, 1]$.

$$\begin{array}{c} X_1 \to \{Y_1\} \\ X_2 \to \{Y_2\} \\ \dots \\ X_k \to \{Y_k\} \\ \hline X \to \{Y\} \end{array} \quad \Leftrightarrow \quad \begin{array}{c} F(X_1) = F(X_1 \cup Y_1) \\ F(X_2) = F(X_2 \cup Y_2) \\ \dots \\ F(X_k) = F(X_k \cup Y_k) \\ \hline F(X) = F(X \cup Y) \end{array}$$

Application-specific semantics (MVDs, CIs, ...):	$\Leftrightarrow$	Corresponding classes of real-valued functions $(\mathcal{F}_{MVDs}, \mathcal{F}_{CIs}, ...)$ and implicit bounds on the order; here: $[2, 2]$.

$$\begin{array}{c} X_1 \to \{Y_1, Z_1\} \\ X_2 \to \{Y_2, Z_2\} \\ \dots \\ X_k \to \{Y_k, Z_k\} \\ \hline X \to \{Y, Z\} \end{array} \quad \Leftrightarrow \quad \begin{array}{c} F(X_1) + F(X_1 \cup Y_1 \cup Z_1) = F(X_1 \cup Y_1) + F(X_1 \cup Z_1) \\ F(X_2) + F(X_2 \cup Y_2 \cup Z_2) = F(X_2 \cup Y_2) + F(X_2 \cup Z_2) \\ \dots \\ F(X_k) + F(X_k \cup Y_k \cup Z_k) = F(X_k \cup Y_k) + F(X_k \cup Z_k) \\ \hline F(X) + F(X \cup Y \cup Z) = F(X \cup Y) + F(X \cup Z) \end{array}$$

FIGURE 2.2. The theoretical framework unifies important implication problems occurring in computer science. Instances of implication problems with application-specific semantics are equivalent to instances of implication problems for additive constraints relative to the corresponding classes of functions.

implication problem is $[2, 2]$-bounded, that is, it only involves disjunctive statements of order exactly 2.

After this syntactic unification, we can also unify the semantics, by considering the implication problem for disjunctive statements relative to the corresponding class of real-functions for the application-specific semantics. For example, for FDs this is the class of Shannon entropy functions, and for conditional independence, this is the class of multi-information functions induced by the class of discrete probability measures. Instances of the implication problem for FDs/CI Statements are then equivalent to implication problems on additive constraints relative to these classes of functions. Figure 2.2 depicts these correspondences for disjunctive statements of order exactly 1 and 2, respectively. Instances of implication problems with the application-specific semantics (e.g., FDs and ARs) are equivalent to instances of implication problems on additive constraints with order exactly 1, relative to the corresponding classes of functions (here: Shannon entropy and frequency functions).

CHAPTER 3

Inference Systems

Given a class of real-valued functions $\mathcal{F}$ and the previously defined (bounded) implication problem relative to $\mathcal{F}$, we would like to determine inference systems that are sound, complete, or both for this implication problem. The usefulness of this approach has already been demonstrated time and again in a wide variety of application areas.

Therefore, in this chapter, we consider inference systems which are sound and/or complete for several well-known implication problems: an inference system that we refer to as system $\mathcal{K}$ (see Figure 3.1) and a generalization of the *semi-graphoid* axioms [**8, 31**] for statements of arbitrary order that we refer to as system $\mathcal{G}$ (see Figure 3.4). These inference systems will serve as *templates* for specialized inference systems in the context of implication problems with bounds on the order of the disjunctive statements. For example, Figure 3.2 depicts inference system $\mathcal{K}$ specialized to the $[2, 2]$-bounded implication problem, and Figure 3.3 depicts the same inference system specialized to the $[1, 1]$-bounded implication problem. Indeed, observe the syntactic similarities with the inference systems shown in Figures 1.4, 1.5, 1.6, and 1.7. Furthermore, note that the inference rule *symmetry* is implicitly sound for disjunctive statements, as the right-hand-side of each disjunctive statement is always a *set* of sets. We will prove these inference systems to be sound with respect to semi-lattice inclusion, that is, we will show that if the semi-lattice of a disjunctive statement is a subset of the union of semi-lattices of a set of CI statements, then the single CI statement can be derived from the set of CI statements. We will be able to harness this property of the inference systems to derive testable properties that imply the soundness and/or completeness of the systems. Before doing so however, we will introduce some basic definitions.

Augmentation

$$\frac{X \rightarrow \mathcal{Y}}{X \cup W \rightarrow \mathcal{Y}}$$

Strong Transitivity

$$\frac{\begin{array}{c} X \rightarrow \mathcal{Y} \cup \mathcal{Z} \\ \forall Z \in \mathcal{Z} : X \cup Z \rightarrow \mathcal{Y} \cup \mathcal{V} \end{array}}{X \rightarrow \mathcal{Y} \cup \mathcal{V}}$$

Composition

$$\frac{\begin{array}{c} X \rightarrow \mathcal{Y} \cup \{Y\} \\ X \rightarrow \mathcal{Y} \cup \{Z\} \end{array}}{X \rightarrow \mathcal{Y} \cup \{Y \cup Z\}}$$

Decomposition

$$\frac{X \rightarrow \mathcal{Y} \cup \{Y \cup Z\}}{X \rightarrow \mathcal{Y} \cup \{Y\}}$$

Triviality

$$\frac{}{X \rightarrow \mathcal{Y} \cup \{\emptyset\}}$$

FIGURE 3.1. Inference system $\mathcal{K}$ which servers as a template for specific instances of inference systems for bounded implication problems.

Augmentation

$$\frac{X \to \{Y, Z\}}{X \cup W \to \{Y, Z\}}$$

Triviality

$$\frac{}{X \to \{Y\} \cup \{\emptyset\}}$$

Strong Transitivity (1)

$$\frac{X \to \{Y\} \cup \{Z\} \qquad X \cup Z \to \{Y\} \cup \{V\}}{X \to \{Y\} \cup \{V\}}$$

Strong Transitivity (2)

$$\frac{X \to \{Z\} \cup \{Z'\} \qquad X \cup Z \to \{V\} \cup \{V'\} \qquad X \cup Z' \to \{V\} \cup \{V'\}}{X \to \{V\} \cup \{V'\}}$$

Composition

$$\frac{X \to \{Y\} \cup \{W\} \qquad X \to \{Y\} \cup \{Z\}}{X \to \{Y\} \cup \{W \cup Z\}}$$

Decomposition

$$\frac{X \to \{Y\} \cup \{W \cup Z\}}{X \to \{Y\} \cup \{Z\}}$$

FIGURE 3.2. Inference system $\mathcal{K}_2$ specialized to the $[2, 2]$-bounded implication problem for disjunctive statements.

Augmentation

$$\frac{X \to \{Y\}}{X \cup W \to \{Y\}}$$

Strong Transitivity

$$\frac{X \to \{Z\} \qquad X \cup Z \to \{V\}}{X \to \{V\}}$$

Composition

$$\frac{X \to \{Y\} \qquad X \to \{Z\}}{X \to \{Y \cup Z\}}$$

Decomposition

$$\frac{X \to \{Y \cup Z\}}{X \to \{Y\}}$$

Triviality

$$\frac{}{X \to \{\emptyset\}}$$

FIGURE 3.3. Inference system $\mathcal{K}_1$ specialized to the $[1,1]$-bounded implication problem for disjunctive statements.

Triviality

$$\frac{}{X \to \mathcal{Y} \cup \{\emptyset\}}$$

Weak Augmentation

$$\frac{X \to \mathcal{Y} \cup \{Y \cup Z\}}{X \cup Z \to \mathcal{Y} \cup \{Y\}}$$

Weak Contraction

$$\frac{\begin{array}{c} X \to \mathcal{Y} \cup \{Y\} \cup \{Z \cup W\} \\ X \cup Y \to \mathcal{Y} \cup \{Z\} \cup \{W\} \end{array}}{X \to \mathcal{Y} \cup \{Y \cup Z\} \cup \{W\}}$$

FIGURE 3.4. Semi-graphoid inference system $\mathcal{G}$ which servers as a template for specific instances of inference systems for bounded implication problems for saturated disjunctive statements.

Triviality

$$\frac{}{X \to \{Y\} \cup \{\emptyset\}}$$

Weak Augmentation

$$\frac{X \to \{Y\} \cup \{W \cup Z\}}{X \cup Z \to \{W\} \cup \{Y\}}$$

Weak Contraction

$$\frac{X \to \{Y\} \cup \{Z \cup W\} \qquad X \cup Y \to \{Z\} \cup \{W\}}{X \to \{Y \cup Z\} \cup \{W\}}$$

FIGURE 3.5. Semi-graphoid inference system $\mathcal{G}$ specialized to the $[2, 2]$-bounded implication problem for saturated disjunctive statements.

1. Preliminaries

DEFINITION 3.1. Let S be a finite set, let $\mathcal{I}$ be an inference system for disjunctive statements, let $\mathcal{C}$ be a set of disjunctive statements over S, and let c be a disjunctive statement over S. We say that c can be *derived* from $\mathcal{C}$ *under* $\mathcal{I}$, if there exists a finite sequence of disjunctive statements $c_1, \ldots, c_n$ such that

(1) for $i = 1, \ldots, n$, c_i is either given (i.e., an element of $\mathcal{C}$) or c_i can be derived from some of the constraints $c_1, \ldots, c_{i-1}$ using an inference rule in $\mathcal{I}$;
(2) $c_n = c$.

The sequence $c_1, \ldots, c_n$ is called a *derivation*.

Consider an $[\ell, u]$-bounded implication problem. Let $(\mathcal{C}, c)$ be an instance of this bounded implication problem over some finite set S. If all disjunctive statements in a derivation of c from $\mathcal{C}$ have order at most u, we say that the derivation is *upper bounded*. If all the disjunctive statements in a derivation of c from $\mathcal{C}$ have order at least ℓ, we say that the derivation is *lower bounded*. Finally, if a derivation is both *lower bounded* and *upper bounded*, we say that it is *bounded*. The bounded derivability of c from $\mathcal{C}$ under $\mathcal{I}$ is denoted by $\mathcal{C} \vdash_{\mathcal{I}} c$. It goes without saying that bounded derivability is very desirable as it restricts the possible intermediate disjunctive statements to a finite number.

EXAMPLE 3.2. Consider the inference system $\mathcal{K}$ in Figure 3.1. Let $S = \{a, b, c, d\}$, let $\mathcal{C} = \{\{a\} \to \{\{b, c\}, \{d\}\}, \{c\} \to \{\{d\}\}\}$, and let c be $\{a, b\} \to \{\{d\}\}$. Observe that all disjunctive statements in $\mathcal{C} \cup \{c\}$ are of order 1 or 2. We can show that $\mathcal{C} \vdash_{\mathcal{K}} c$ by exhibiting a derivation:

(1)	$\{c\}$	$\to$	$\{\{d\}\}$	(given)
(2)	$\{a, b, c\}$	$\to$	$\{\{d\}\}$	(**augmentation**)
(3)	$\{a\}$	$\to$	$\{\{b, c\}, \{d\}\}$	(given)
(4)	$\{a\}$	$\to$	$\{\{c\}, \{d\}\}$	(**decomposition**)
(5)	$\{a, b\}$	$\to$	$\{\{c\}, \{d\}\}$	(**augmentation**)
(6)	$\{a, b\}$	$\to$	$\{\{d\}\}$	(**strong transitivity** on (5) and (2)), with $X = \{a, b\}, \mathcal{Y} = \{\{d\}\}, \mathcal{V} = \emptyset,$ and $\mathcal{Z} = \{\{c\}\}$).

All disjunctive statements in the above derivation have order 1 or 2. We may therefore conclude that this derivation is *bounded*.

Notice that the only disjunctive statements that can be inferred from the empty set, that is, the set containing no disjunctive statements, are precisely those that satisfy the triviality rule. We shall therefore call these disjunctive statements *trivial*.

2. Semi-Lattice Characterization of Inference Systems

Based on the notion of bounded derivability, we prove that (i) $\mathcal{K}$ is sound and complete with respect to *semi-lattice inclusion* for disjunctive statements, and (ii) $\mathcal{G}$ is sound and complete with respect to *semi-lattice inclusion* for saturated disjunctive statements. To prove this, we will first need to introduce the notion of a *witness set*.

Definition 3.3 (Witness set). Let S be a finite set and $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$ be a set of subsets of S. The set $\mathcal{W}(\mathcal{Y})$ of all witness sets of $\mathcal{Y}$ is defined by

$$\mathcal{W}(\mathcal{Y}) = \{\{w_1, \ldots, w_n\} \mid w_1 \in Y_1 \wedge \ldots \wedge w_n \in Y_n\}.$$

Every element in $\mathcal{W}(\mathcal{Y})$ is called a *witness set*. Notice that the size of a witness set always equals the size of $\mathcal{Y}$. Furthermore, we have that $\mathcal{W}(\emptyset) = \{\emptyset\}$ and $\mathcal{W}(\mathcal{Y}) = \emptyset$ if $\emptyset \in \mathcal{Y}$.

EXAMPLE 3.4. Let $S = \{a, b, c, d\}$. Then,

$$\mathcal{W}(\{\{b, c\}, \{d\}\}) = \{\{b, d\}, \{c, d\}\};$$
$$\mathcal{W}(\{\{a, b\}, \{c, d\}\}) = \{\{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}\};$$
$$\mathcal{W}(\{\{a\}, \{c\}\}) = \{\{a, c\}\};$$
$$\mathcal{W}(\{\{a, b\}, \emptyset\}) = \emptyset; \text{ and}$$
$$\mathcal{W}(\emptyset) = \{\emptyset\}.$$

We will now show that the semi-lattice decomposition of a disjunctive statement can be equivalently expressed using lattices induced by the witness sets of the disjunctive statements.

PROPOSITION 3.5. *Let S be a finite set, let $X \subseteq S$, and let $\mathcal{Y}$ a set of subsets of S. Then,*

$$\mathcal{L}(X, \mathcal{Y}) = \bigcup_{W \in \mathcal{W}(\mathcal{Y})} [X, \overline{W}].$$

Proof: We first show that $\mathcal{L}(X, \mathcal{Y}) \subseteq \bigcup_{W \in \mathcal{W}(\mathcal{Y})} [X, \overline{W}]$. Let $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$, $U \in \mathcal{L}(X, \mathcal{Y})$, and $\mathcal{Y}' = \{Y_1 - U, \ldots, Y_n - U\}$. Note that $U \supseteq X$, and that every element of $\mathcal{Y}'$ is non-empty since $U \subseteq Y$ for all $Y \in \mathcal{Y}$. Now, for $W \in \mathcal{W}(\mathcal{Y}')$ we have that $U \subseteq \overline{W}$. Thus, $U \in [X, \overline{W}]$. It now suffices to observe that W is also a witness set of $\mathcal{Y}$.

To show the converse inclusion, let $U \in [X, \overline{W}]$ for some witness set W of $\mathcal{Y}$. Since $U \cap W = \emptyset$, it follows that $U \not\supseteq Y$ for all $Y \in \mathcal{Y}$, and, therefore, $U \in \mathcal{L}(X, \mathcal{Y})$.

EXAMPLE 3.6. Let $S = \{a, b, c, d\}$ be a finite set. Then,

$$\begin{aligned}
\mathcal{L}(\{a\}, \{\{b,c\},\{d\}\}) &= [\{a\},\{a,b,c,d\}] - ([\{b,c\},\{a,b,c,d\}] \cup [\{d\},\{a,b,c,d\}]) \\
&= \{\{a\},\{a,b\},\{a,c\}\} = [\{a\},\{a,b\}] \cup [\{a\},\{a,c\}] \\
&= [\{a\},\overline{\{c,d\}}] \cup [\{a\},\overline{\{b,d\}}] = \textstyle\bigcup_{W\in\mathcal{W}(\{\{b,c\},\{d\}\})}[\{a\},\overline{W}]; \\
\mathcal{L}(\emptyset, \{\{a,b\},\{c,d\}\}) &= [\emptyset,\{a,b,c,d\}] - ([\{a,b\},\{a,b,c,d\}] \cup [\{c,d\},\{a,b,c,d\}]) \\
&= \{\emptyset,\{a\},\{b\},\{c\},\{d\},\{a,c\},\{a,d\},\{b,c\},\{b,d\}\} \\
&= [\emptyset,\{b,d\}] \cup [\emptyset,\{b,c\}] \cup [\emptyset,\{a,d\}] \cup [\emptyset,\{a,c\}] \\
&= [\emptyset,\overline{\{a,c\}}] \cup [\emptyset,\overline{\{a,d\}}] \cup [\emptyset,\overline{\{b,c\}}] \cup [\emptyset,\overline{\{b,d\}}] \\
&= \textstyle\bigcup_{W\in\mathcal{W}(\{\{a,b\},\{c,d\}\})}[\emptyset,\overline{W}]; \text{ and} \\
\mathcal{L}(\{a\}, \{\{b,c\},\emptyset\}) &= [\{a\},\{a,b,c,d\}] - ([\{b,c\},\{a,b,c,d\}] \cup [\emptyset,\{a,b,c,d\}]) \\
&= \emptyset = \textstyle\bigcup_{W\in\mathcal{W}(\{\{b,c\},\emptyset\})}[\{a\},\overline{W}].
\end{aligned}$$

The following proposition states that inference system $\mathcal{K}$ is *sound* with respect to bounded semi-lattice inclusion, that is, when some of the inference rules of inference system $\mathcal{K}$ are applied to a set of disjunctive statements $\mathcal{C}$ to derive a single disjunctive statement c, then $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(c)$.

PROPOSITION 3.7. *Let S be a finite set, let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements under consideration, let $\mathcal{C}$ be a set of disjunctive statements, and let c be a single disjunctive statement, all over S and satisfying the given bounds. If $\mathcal{C} \vdash_{\mathcal{K}} c$, then $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(c)$.*

Proof: It suffices to show that, for each inference rule of system $\mathcal{K}$, the semi-lattice decomposition of the consequence of the inference rule is a subset of the union of the semi-lattice decompositions of the antecedents of the inference rule. We start by showing this for **strong transitivity** (see Figure 3.1). Thus, let $U \in \mathcal{L}(X, \mathcal{Y} \cup \mathcal{V})$. By Definition 2.6, $U \supseteq X$. We distinguish two cases. *Case 1*: There exists Z in $\mathcal{Z}$ such that $U \supseteq Z$. Then $U \supseteq X \cup Z$ and, therefore, $U \in \mathcal{L}(X \cup Z, \mathcal{Y} \cup \mathcal{V})$. *Case 2*: There does not exist $Z \in \mathcal{Z}$ such that $U \supseteq Z$. Then, $U \in \mathcal{L}(X, \mathcal{Y} \cup \mathcal{Z})$.

Consider inference rule **augmentation**. Thus, let $U \in \mathcal{L}(X \cup W, \mathcal{Y})$. By Definition 2.6, $U \supseteq X \cup W$ and for all $Y \in \mathcal{Y}$ we have that $U \not\supseteq Y$. But then we also have that $U \supseteq X$ and for all $Y \in \mathcal{Y}$ we have that $U \not\supseteq Y$. Hence, $U \in \mathcal{L}(X, \mathcal{Y})$.

Consider inference rule **decomposition**. Thus, let $U \in \mathcal{L}(X, \mathcal{Y} \cup \{Y\})$. By Definition 2.6, $U \supseteq X$, $Y \not\subseteq U$ and for all $V \in \mathcal{Y}$ we have that $U \not\supseteq V$. But then we also have that $U \supseteq X$, $Y \cup Z \not\subseteq U$, and for all $V \in \mathcal{Y}$ we have that $U \not\supseteq V$. Hence, $U \in \mathcal{L}(X, \mathcal{Y} \cup \{Y \cup Z\})$.

Finally, consider inference rule **composition**. Thus, let $U \in \mathcal{L}(X, \mathcal{Y} \cup \{Y \cup Z\})$. By Definition 2.6, $U \supseteq X$, $Y \cup Z \not\subseteq U$, and for all $V \in \mathcal{Y}$ we have that $U \not\supseteq V$. We distinguish two cases. *Case 1*: We have that $Y \subseteq U$. Then $Z \not\supseteq U$ because otherwise we would have that $Y \cup Z \subseteq U$. Hence, $U \in \mathcal{L}(X, \mathcal{Y} \cup \{Z\})$. *Case 2*: We have that $Y \not\subseteq U$. Then, $U \in \mathcal{L}(X, \mathcal{Y} \cup \{Y\})$.

We proceed with showing that inference system $\mathcal{K}$ is also complete with respect to bounded semi-lattice inclusion. Therefore, we first define the *witness decomposition* of a disjunctive statement.

DEFINITION 3.8 (Witness decomposition). Let S be a finite set. The *witness decomposition* of the disjunctive statement $X \to \mathcal{Y}$ over S is defined by

$$\texttt{wdec}(X \to \mathcal{Y}) := \{X \to \mathcal{A}(W) \mid W \in \mathcal{W}(\mathcal{Y})\},$$

with $\mathcal{A}(W) = \{\{w\} \mid w \in W\}$.

The following proposition states that a witness decomposition of a disjunctive statement is a normal form of the disjunctive statement relative to both the semi-lattice inclusion and the inference system $\mathcal{K}$, that is, one representation can be derived from the other using inference rules from $\mathcal{K}$, and both representations have the same associated semi-lattice.

PROPOSITION 3.9. *Let c be a disjunctive statement over some finite set S satisfying the bounds of the implication problem under consideration. Then (1) $\{c\} \vdash_{\mathcal{K}} c'$ for each c' in* $\texttt{wdec}(c)$*; (2)* $\texttt{wdec}(c) \vdash_{\mathcal{K}} c$*; and (3)* $\mathcal{L}(c) = \mathcal{L}(\texttt{wdec}(c))$.

Proof: To prove the first statement, let $c = X \to \mathcal{Y}$ and $X \to \mathcal{A}(W) \in \texttt{wdec}(c)$. Then $X \to \mathcal{A}(W)$ can be derived from $X \to \mathcal{Y}$ by applications of the **decomposition** rule. Hence, $\{c\} \vdash_{\mathcal{K}} c'$.

By Definition 3.8, we know that, for every $W \in \mathcal{W}(\mathcal{Y})$, we have that $X \to \mathcal{A}(W) \in \texttt{wdec}(c)$. Let $Y \in \mathcal{Y}$. We know that $\mathcal{W}(\mathcal{Y}-\{Y\})\cup\{\{y\}\}$ is an element of $\mathcal{W}(\mathcal{Y})$, for all $y \in Y$, and, therefore, $X \to \mathcal{W}(\mathcal{Y} - \{Y\}) \cup \{\{y\}\} \in \texttt{wdec}(c)$, for all $y \in Y$. Now, by repeatedly applying **composition**, we can infer the disjunctive statements $X \to \mathcal{W}(\mathcal{Y} - \{Y\}) \cup \{Y\}$, for all $Y \in \mathcal{Y}$. The process of repeatedly applying **composition** can be continued to these statements to finally derive the disjunctive statement $X \to \mathcal{Y}$. Hence, $\texttt{wdec}(c) \vdash_{\mathcal{K}} c$.

To prove the third statement, let $X \to \mathcal{A}(W) \in \texttt{wdec}(c)$. Then $\mathcal{L}(X, \mathcal{A}(W)) = [X, \overline{W}]$. The statement now follows directly from Definition 3.8 and Proposition 3.5.

We are now ready to prove our first main result, that is, the soundness and completeness of inference system $\mathcal{K}$ with respect to semi-lattice inclusion.

Theorem 3.10. *Let $\mathcal{C}$ be a set of disjunctive statements, and let c be a disjunctive statement, all over some finite set S and satisfying the bounds of the implication problem under consideration. Then $\mathcal{C} \vdash_{\mathcal{K}} c$ if and only if $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(c)$.*

Proof: The "only if" was already shown in Proposition 3.7, so we now turn to the "if."

Let us denote $\texttt{wdec}(\mathcal{C}) = \bigcup_{c' \in \mathcal{C}} \texttt{wdec}(c')$ and suppose $c = X \to \mathcal{Y}$. Let $X \to \mathcal{A}(W) \in \texttt{wdec}(c)$ with $W = \{w_1, \ldots, w_n\}$. From the assumption $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(c)$ and Proposition 3.9(3), it follows that $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(X, \mathcal{A}(W))$ (1). By Proposition 3.9(1), it suffices to show that $\texttt{wdec}(\mathcal{C}) \vdash_{\mathcal{K}} X \to \mathcal{A}(W)$. In order to achieve this, we prove the stronger statement

$$\forall V \in [X, \overline{W}] : \texttt{wdec}(\mathcal{C}) \vdash_{\mathcal{K}} V \to \mathcal{A}(W)$$

by downward induction on the lattice $[X, \overline{W}]$.

For the base case, we need to show that $\texttt{wdec}(\mathcal{C}) \vdash_{\mathcal{K}} \overline{W} \to \mathcal{A}(W)$. By (1), $\overline{W}$ is in $\mathcal{L}(\mathcal{C})$. Hence, by Proposition 3.9, (1), there exists a disjunctive statement $X' \to \mathcal{A}(W') \in \texttt{wdec}(\mathcal{C})$ such that $\overline{W} \in \mathcal{L}(X', \mathcal{A}(W'))$. Note that $W' \subseteq W$. If $W' \neq W$, we first derive $X' \to \mathcal{A}(W)$

from $X' \to \mathcal{A}(W')$ using **strong transitivity**. Next, we derive $\overline{W} \to \mathcal{A}(W)$ from $X' \to \mathcal{A}(W')$ by **augmentation**.

For the induction step, let $X \subseteq V \subset \overline{W}$. The induction hypothesis states that, for all V' with $V \subset V' \subseteq \overline{W}$, $\texttt{wdec}(\mathcal{C}) \vdash_{\mathcal{K}} V' \to \mathcal{A}(W)$. By (1), V is in $\mathcal{L}(\mathcal{C})$. Hence, by Proposition 3.9,(1), there exists a disjunctive statement $X' \to \mathcal{A}(W') \in \texttt{wdec}(\mathcal{C})$ such that $V \in \mathcal{L}(X', \mathcal{A}(W'))$. Since $X' \subseteq V$, we can use **augmentation** to derive $V \to \mathcal{A}(W')$. We may write $V \to \mathcal{A}(W')$ as $V \to \mathcal{A}(W' \cap W) \cup \mathcal{A}(W' \cap \overline{W})$ (a). Furthermore, by the induction hypothesis and $W' \cap V = \emptyset$, we have, for all w' in $W' \cap \overline{W}$, that $\texttt{wdec}(\mathcal{C}) \vdash_{\mathcal{K}} V \cup \{w'\} \to \mathcal{A}(W)$ (b). Now, by applying **strong transitivity** to (a) and the statements (b), we can finally infer $V \to \mathcal{A}(W)$.

Please note that the above proof provides an algorithm to construct the derivation for any given instance of the implication problem. We will now use the semi-lattice inclusion property of inference system $\mathcal{K}$ to derive properties of classes of real-valued functions that imply the soundness and completeness of the system.

3. Semantic Properties of Inference Systems

So far, as we have unified a large class of implication problems by reducing them to instances of bounded implication problems on additive constraints on real-valued functions. We will now derive properties of classes of real-valued functions that guarantee soundness and completeness or both of the two inference systems introduced in this chapter. In particular, if we could succeed in deriving *testable* properties of classes of real-valued functions that imply soundness and completeness properties, the framework would allow us to approach numerous implication problems from a new angle of attack.

3.1. Soundness. First, we define the notion of *bounded soundness*. Of course, we want the theoretical framework to be applicable to existing important applications. Most of these applications restrict the order of the disjunctive statements under consideration. For example, since multivalued dependencies can be interpreted as saturated disjunctive statements of order exactly 2, we want to be able to generally restrict the corresponding implication problems and inference systems to disjunctive statements of order exactly 2.

Therefore, the discussion of the inference systems and their semantic properties will be in the context of the $[\ell, u]$-bounded implication problem relative to some class $\mathcal{F}$ of real-valued functions, for two natural numbers u and ℓ with $1 \leq \ell \leq u$.

Let $\mathcal{I}$ be an inference system for disjunctive statements. We say that $\mathcal{I}$ is $[\ell, u]$-*sound* relative to $\mathcal{F}$ if, for each finite set S, for each set $\mathcal{C}$ of disjunctive statements over S, and for each single disjunctive statement C over S, all of order at least ℓ and at most u, we have that $\mathcal{C} \vdash_{\mathcal{I}} c$ implies $\mathcal{C} \models_{\mathcal{F}_S} c$.

Notice that $\mathcal{K}$ is a "stronger" inference system than $\mathcal{G}$ in the sense that every derivation under the inference system $\mathcal{G}$ is also achievable under the inference system $\mathcal{K}$.

LEMMA 3.11. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$. If $\mathcal{K}$ is $[\ell, u]$-sound relative to $\mathcal{F}$, then $\mathcal{G}$ is $[\ell, u]$-sound relative to $\mathcal{F}$.*

In order to characterize bounded soundness of the two inference systems under consideration relative to a class of real-valued functions, we introduce the zero-density property for such classes.

DEFINITION 3.12 (Zero density). Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. We say that $\mathcal{F}$ has the $[\ell, u]$-*zero-density property* if, for each finite set S, for each real-valued function F over S in $\mathcal{F}_S$, and for each disjunctive statement c over S of order at least ℓ and at most u, we have that F satisfies c implies $\Delta F(U) = 0$, for all $U \in \mathcal{L}(c)$.

We can now characterize *bounded soundness* of the inference system $\mathcal{K}$.

THEOREM 3.13. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. Then the following statements are equivalent:*

(1) *Augmentation and decomposition are $[\ell, u]$-sound relative to $\mathcal{F}$;*
(2) *$\mathcal{F}$ has the $[\ell, u]$-zero-density property; and*
(3) *$\mathcal{K}$ is $[\ell, u]$-sound relative to $\mathcal{F}$.*

Proof: We first prove that (1) implies (2). Assume that **augmentation** and **decomposition** are $[\ell, u]$-*sound* relative to $\mathcal{F}$. Let S be a finite set, let F be in $\mathcal{F}_S$, let $X \to \mathcal{Y}$ be a disjunctive statement over S of order at least ℓ and at most u, and assume that F satisfies $X \to \mathcal{Y}$, i.e., $\Delta^{\mathcal{Y}} F(X) = 0$. We need to show that $\Delta F(V) = 0$ for each $V \in \mathcal{L}(X, \mathcal{Y})$. The proof goes by downward induction on the structure of the semi-lattice decomposition.

First, we observe that, by Proposition 3.5, $\mathcal{L}(X, \mathcal{Y}) = \bigcup_{W \in \mathcal{W}(\mathcal{Y})} [X, \overline{W}]$. Hence, for the base case, we must prove that $\Delta F(\overline{W}) = 0$ for all W in $\mathcal{W}(\mathcal{Y})$. Since **decomposition** is $[\ell, u]$-*sound* relative to $\mathcal{F}$, it follows that F also satisfies $X \to \mathcal{A}(W)$. Furthermore, since **augmentation** is $[\ell, u]$-*sound* relative to $\mathcal{F}$, and $X \subseteq \overline{W}$, it follows that F also satisfies $\overline{W} \to \mathcal{A}(W)$, i.e., $\Delta^{\mathcal{A}(W)} F(\overline{W}) = 0$. Now, by Proposition 2.9, $\Delta^{\mathcal{A}(W)} F(\overline{W}) = \Delta F(\overline{W}) = 0$.

We now turn to the induction step. Let $V \in \mathcal{L}(X, \mathcal{Y})$. The induction hypothesis states that $\Delta F(U) = 0$ for all U in $\mathcal{L}(X, \mathcal{Y})$ that are strict supersets of V. By Proposition 3.5, there exists W in $\mathcal{W}(\mathcal{Y})$ with $V \subseteq \overline{W}$. As in the base case, we can apply **decomposition** and **augmentation** to derive that F satisfies $V \to \mathcal{A}(W)$. Hence, $\Delta^{\mathcal{A}(W)} F(V) = \sum_{U \in \mathcal{L}(V, \mathcal{A}(W))} \Delta F(U) = \Delta F(V) = 0$, since all other densities in the sum are zero by the induction hypothesis.

We next prove that (2) implies (3). Let S be a finite set, let $\mathcal{C}$ be a set of disjunctive statements over S, and let c be a single disjunctive statement over S, all of order at least ℓ and at most u. Assume that $\mathcal{C} \vdash_{\mathcal{K}} c$, which, by Theorem 3.10, is equivalent to $\mathcal{L}(c) \subseteq \mathcal{L}(\mathcal{C})$. Now, let F be any function in $\mathcal{F}_S$ that satisfies all disjunctive statements in $\mathcal{C}$. Let U be in $\mathcal{L}(c)$. Then there is a disjunctive statement c' in $\mathcal{C}$ such that U is in $\mathcal{L}(c')$. Since F satisfies c', and $\mathcal{F}$ has the $[\ell, u]$-zero-density property, it follows that $\Delta F(U) = 0$. Let $c = X \to \mathcal{Y}$. Since $\Delta^{\mathcal{Y}} F(X) = \sum_{U \in \mathcal{L}(c)} \Delta F(U) = 0$, we have that F satisfies $c = X \to \mathcal{Y}$. We may thus conclude that $\mathcal{C} \models_{\mathcal{F}} c$.

Finally, (3) trivially implies (1) since **augmentation** and **decomposition** are inference rules of $\mathcal{K}$.

By Lemma 3.11, conditions (1) and (2) of Theorem 3.13 each imply that also $\mathcal{G}$ is $[\ell, u]$-sound relative to S.

When only saturated statements are considered, a result analogous to Theorem 3.13 holds.

THEOREM 3.14. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. Then the following statements are equivalent:*

(1) *Weak augmentation is $[\ell, u]$-sound relative to $\mathcal{F}$ for saturated disjunctive statements;*
(2) *$\mathcal{F}$ has the $[\ell, u]$-zero-density property for saturated disjunctive statements;*
(3) *$\mathcal{K}$ is $[\ell, u]$-sound relative to $\mathcal{F}$ for saturated disjunctive statements; and*
(4) *$\mathcal{G}$ is $[\ell, u]$-sound relative to $\mathcal{F}$ for saturated disjunctive statements.*

Proof: We first prove that (1) implies (2). Assume that **weak augmentation** is $[\ell, u]$-*sound* relative to $\mathcal{F}$ for saturated disjunctive statements. Let S be a finite set, let F be in $\mathcal{F}_S$, let $X \to \mathcal{Y}$ be a saturated disjunctive statement over S of order at least ℓ and at most u, and assume that F satisfies $X \to \mathcal{Y}$, i.e., $\Delta^{\mathcal{Y}} F(X) = 0$. We need to show that $\Delta F(V) = 0$ for each $V \in \mathcal{L}(X, \mathcal{Y})$. The proof goes by downward induction on the structure of the semi-lattice decomposition.

First, we observe that, by Proposition 3.5, $\mathcal{L}(X, \mathcal{Y}) = \bigcup_{W \in \mathcal{W}(\mathcal{Y})} [X, \overline{W}]$. Hence, for the base case, we must prove that $\Delta F(\overline{W}) = 0$ for all W in $\mathcal{W}(\mathcal{Y})$. Since **weak augmentation** is $[\ell, u]$-*sound* relative to $\mathcal{F}$ for saturated disjunctive statements, it follows that F also satisfies $\overline{W} \to \mathcal{A}(W)$, that is, $\Delta^{\mathcal{A}(W)} F(\overline{W}) = 0$. Now, by Proposition 2.9, $\Delta^{\mathcal{A}(W)} F(\overline{W}) = \Delta F(\overline{W}) = 0$.

We now turn to the induction step. Let $V \in \mathcal{L}(X, \mathcal{Y})$. The induction hypothesis states that $\Delta F(U) = 0$ for all U in $\mathcal{L}(X, \mathcal{Y})$ that are strict supersets of V. By Proposition 3.5, there exists W in $\mathcal{W}(\mathcal{Y})$ with $V \subseteq \overline{W}$. As in the base case, we can apply **weak augmentation** to derive that F satisfies $V \to \mathcal{A}(W)$. Hence, $\Delta^{\mathcal{A}(W)} F(V) = \sum_{U \in \mathcal{L}(V, \mathcal{A}(W))} \Delta F(U) = \Delta F(V) = 0$, since all other densities in the sum are zero by the induction hypothesis (all elements $U \in \mathcal{L}(V, \mathcal{A}(W))$ are elements in $\mathcal{L}(X, \mathcal{Y})$ and $U \supset V$).

We next prove that (2) implies (3). Let S be a finite set, let $\mathcal{C}$ be a set of saturated disjunctive statements over S, and let c be a single saturated disjunctive statement over S, all of order at least ℓ and at most u. Assume that $\mathcal{C} \vdash_{\mathcal{K}} c$, which, by Theorem 3.10, is equivalent to $\mathcal{L}(c) \subseteq \mathcal{L}(\mathcal{C})$. Now, let F be any function in $\mathcal{F}_S$ that satisfies all saturated disjunctive statements in $\mathcal{C}$. Let U be in $\mathcal{L}(c)$. Then there exists a saturated disjunctive statement c' in $\mathcal{C}$ such that U is in $\mathcal{L}(c')$. Since F satisfies c', and $\mathcal{F}$ has the $[\ell, u]$-zero-density property, it follows that $\Delta F(U) = 0$. Let $c = X \rightarrow \mathcal{Y}$ be a saturated disjunctive statement. Since $\Delta^{\mathcal{Y}} F(X) = \sum_{U \in \mathcal{L}(c)} \Delta F(U) = 0$, we have that F satisfies $c = X \rightarrow \mathcal{Y}$. We may thus conclude that $\mathcal{C} \models_{\mathcal{F}} c$.

Of course, (3) implies (4) by Lemma 3.11.

Finally, (4) trivially implies (1) since **weak augmentation** is an inference rules of $\mathcal{G}$.

The importance of this results stems from the fact that it is often not difficult to test whether the inference rules augmentation and decomposition are sound. In fact, there are many important cases where this is already established (see the following example). In addition, the zero-density property holds for many classes of measures. We refer the reader to Sayrafi's dissertation [**36**] for an in-depth discussion of these classes of real-valued functions.

Example 3.15. By Theorem 3.13 we have that $\mathcal{K}$ and $\mathcal{G}$ are $[1, 1]$-sound for the implication problem of functional dependencies, because augmentation and decomposition are sound for functional dependencies. In addition, $\mathcal{K}$ and $\mathcal{G}$ are $[1, u]$-sound for generalized disjunctive rules in frequent pattern mining and for implication formulae in propositional logic, for any upper bound u, because augmentation and decomposition are $[1, u]$-sound in both cases.

By Theorem 3.14, $\mathcal{G}$ is $[2, 2]$-sound for both the implication problem for multivalued dependencies and the implication problem for saturated conditional independence statements, because weak augmentation is $[2, 2]$-sound in both cases. Hence, by the same theorem, we have that the respective classes of real-valued functions have the $[2, 2]$-zero-density property, and we can conclude that $\mathcal{K}$ is also $[2, 2]$-sound in both cases by invoking Theorem 3.13.

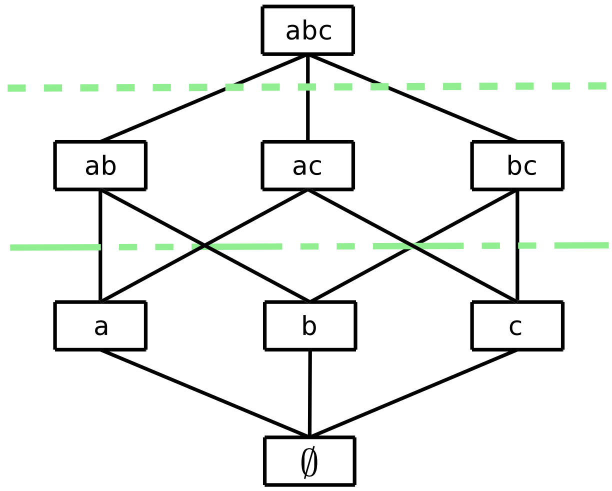

FIGURE 3.6. Hasse diagram of the lattice $[\emptyset, \{a, b, c\}]$. The elements below the topmost dashed line constitute the set $\mathcal{S}^1(S) = \{\emptyset, a, b, c, ab, ac, bc\}$ and the elements below the second dashed line constitute the set $\mathcal{S}^2(S) = \{\emptyset, a, b, c\}$.

3.2. Completeness. First, we formally define the notion of *bounded completeness* of inference rules and inference systems in our framework. Let $\mathcal{I}$ be an inference system for disjunctive statements and let $\ell, u \in \mathbf{N}$ with $\ell \leq u$. We say that $\mathcal{I}$ is $[\ell, u]$-*sound* relative to $\mathcal{F}$ if, for each finite set S, for each set $\mathcal{C}$ of disjunctive statements over S, and for each single disjunctive statement of S, all of order at least ℓ and at most u, we have that $\mathcal{C} \vdash_{\mathcal{F}_S} c$ implies $\mathcal{C} \models_{\mathcal{I}} c$. The counterpart to Lemma 3.11 for bounded completeness is as follows.

LEMMA 3.16. *If $\mathcal{G}$ is $[\ell, u]$-complete relative to $\mathcal{F}$, then $\mathcal{K}$ is $[\ell, u]$-complete relative to $\mathcal{F}$.*

We now turn to inference system $\mathcal{K}$ and derive properties that imply its completeness.

DEFINITION 3.17 (Kronecker density). Let S be a finite set and let $V \subseteq S$. The *Kronecker density* of V, denoted δ_V, is the function from 2^S into the reals for which $\delta_V(V) = 1$ and $\delta_V(X) = 0$ if $X \neq V$. The *Kronecker-induced function* of V, denoted F_V, is the function whose density is the *Kronecker density* of V, i.e., $F_V(X) = \sum_{X \subseteq U \subseteq S} \delta_V(U)$, for $X \subseteq S$.

We now define a property on classes of real-valued functions that guarantees completeness of the inference system $\mathcal{K}$.

DEFINITION 3.18 (Kronecker property). Let S be a finite set, $\mathcal{F}_S$ a set of real-valued functions over S, and $\mathcal{V}$ a set of subsets of S. We say that $\mathcal{F}_S$ has the *Kronecker property* on $\mathcal{V}$ if, for each U in $\mathcal{V}$, there exists a nonzero real number c_U and a set $D_U = \{d_{U,V} \in \mathbf{R} \mid V \notin \mathcal{V}\}$[1] such that

$$F_{\mathcal{V},U,c_U,D_U} = c_U F_U + \sum_{V \subseteq S,\, V \notin \mathcal{V}} d_{U,V} F_V$$

is in $\mathcal{F}$.

Note that, for all X in $\mathcal{V}$, $\Delta F_{\mathcal{V},U,c_U,D_U}(X) = c_U$ if $X = U$, and $\Delta F_{\mathcal{V},U,c_U,D_U}(X) = 0$ if $X \neq U$.

For some number i, let $\mathcal{S}^i(S)$ denote the set of all subsets of S of size at most $|S| - i$. Figure 3.6 depicts the sets $\mathcal{S}^i(S)$ for $S = \{a, b, c\}$ and $i = 0, ..., 2$. We now show how the Kronecker property on $\mathcal{S}^i$ relates to the bounded completeness of $\mathcal{K}$. The following example is intended to illustrate the zero-density and Kronecker properties on two different classes of functions.

EXAMPLE 3.19. Let $S = \{a, b, c\}$, let $\mathcal{F}_1 = \{F_\emptyset, F_a, F_b, F_c\}$, and let $\mathcal{F}_2 = \{F_x\}$, where the densities for each real-valued function are given by the table in Figure 3.7. The densities of the remaining subsets of S are assumed to be 0 for each function. Now, $\mathcal{S}^2(S) = \{\emptyset, a, b, c\}$ and, therefore, $\mathcal{F}_1$ has the Kronecker property on $\mathcal{S}^2(S)$ since $F_{\mathcal{S}^2(S),U,c_U,D_U} = F_U$ for all $U \in \mathcal{S}^2(S)$, and the zero-density property. $\mathcal{F}_2$ does not have the Kronecker property. It also does not have the zero-density property as there exists a real-valued function in $\mathcal{F}_2$ (namely F_x) that satisfies $\emptyset \to \{\{b\}, \{c\}\}$ while $\Delta F_x(\emptyset) \neq 0$.

[1]The real numbers $d_{U,V}$ can be zero.

	$\emptyset$	$\{a\}$	$\{b\}$	$\{c\}$
$\Delta F_\emptyset$	$+0.1$	0	0	0
ΔF_a	0	-0.3	0	0
ΔF_b	0	0	-0.6	0
ΔF_c	0	0	0	$+0.9$
ΔF_x	-0.2	$+0.2$	$+0.6$	$+0.3$

FIGURE 3.7. Densities of several real-valued functions.

THEOREM 3.20. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. If, for every finite set S, $\mathcal{F}_S$ has the Kronecker property on $\mathcal{S}^\ell(S)$, then $\mathcal{K}$ is $[\ell, u]$-complete relative to $\mathcal{F}$.*

Proof: Assume to the contrary that $\mathcal{K}$ is not $[\ell, u]$-complete relative to $\mathcal{F}$. Then, there exists a finite set S, a set $\mathcal{C}$ of disjunctive statements over S, and a single disjunctive statement c over S, all of order at least ℓ and at most u, such that $\mathcal{C} \models_{\mathcal{F}_S} c$ but $\mathcal{C} \not\vdash_{\mathcal{K}} X \to \mathcal{Y}$, or, equivalently, by Theorem 3.10, $\mathcal{L}(c) \nsubseteq \mathcal{L}(\mathcal{C})$. Choose U in $\mathcal{L}(c) - \mathcal{L}(\mathcal{C})$. By Definition 3.3 and Proposition 3.5, $U \in \mathcal{S}^\ell(S)$. Since $\mathcal{F}_S$ has the Kronecker property on $\mathcal{S}^\ell(S)$, there exists a nonzero real number c_U, and a set $D_U = \{d_{U,V} \in \mathbf{R} \mid V \notin \mathcal{S}^\ell(S)\}$ such that $F_{\mathcal{S}^\ell(S),U,c_U,D_U} \in \mathcal{F}_S$. By Definition 3.18, $\Delta F_{\mathcal{S}^\ell(S),U,c_U,D_U}(U) = c_U \neq 0$ and $\Delta F_{\mathcal{S}^\ell(S),U,c_U,D_U}(W) = 0$ for all other W in $\mathcal{S}^\ell(S)$. From Proposition 2.9, it now follows that $F_{\mathcal{S}^\ell(S),U,c_U,D_U}$ satisfies all disjunctive statements in $\mathcal{C}$, but does *not* satisfy c, contradicting $\mathcal{C} \models_{\mathcal{F}} c$.

EXAMPLE 3.21. By Theorem 3.20, we have that $\mathcal{K}$ is $[1, 1]$-complete for the implication problem for functional dependencies because the class of Shannon entropy functions over a relation schema S has the Kronecker property on $\mathcal{S}^1(S)$. Inference system $\mathcal{K}$ is also $[1, u]$-complete for generalized disjunctive rules in frequent pattern mining, for any upper bound u, because the class of all support functions over a set of items S has the Kronecker property on $\mathcal{S}^1(S)$. In addition, inference system $\mathcal{K}$ is $[1, 1]$-complete for the implication problem on association rules and disjunctive rules in data mining, because the class of support functions over a set of items S has the Kronecker property on $\mathcal{S}^1(S)$ and, therefore, also on $\mathcal{S}^2(S)$.

EXAMPLE 3.22. Combining the results from Examples 3.15 and 3.21 we have that inference system $\mathcal{K}_1$ in Figure 3.3 is sound and complete for the implication problem for functional dependencies and association rules. Furthermore, inference system $\mathcal{K}_2$ in Figure 3.2 is sound and complete for the implication problem for disjunctive rules in data mining.

Notice that, whenever a set of real-valued function over some finite set S has the Kronecker property on $\mathcal{S}^\ell(S)$ for some lower bound $\ell \geq 0$, then it also has the Kronecker property on $\mathcal{S}^i(S)$, for all i with $\ell \leq i$. Hence, for a sequence of bounds $0 \leq \ell \leq \ell' \leq u' \leq u$, we have that the $[\ell, u]$-completeness relative to some class of functions of $\mathcal{K}$ implies also its $[\ell', u']$-completeness.

Theorems 3.20 only works in one direction, unfortunately. In the context of bounded soundness, however, we can also state the converse.

PROPOSITION 3.23. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions.*

(1) *If $\ell \geq 1$ and $\mathcal{K}$ is $[\ell, u]$-sound and $[\ell, u]$-complete relative to $\mathcal{F}$, then, for each finite set S, $\mathcal{F}_S$ has the Kronecker property on $\mathcal{S}^\ell(S)$.*
(2) *If $\ell \geq 2$ and $\mathcal{K}$ is $[\ell, u]$-sound and $[\ell, u]$-complete relative to $\mathcal{F}$ for saturated disjunctive statements, then $\mathcal{F}$ has the Kronecker property on $\mathcal{S}^\ell(S)$.*

Proof: We start with (2). Thus, let $\ell \geq 2$, and assume that $\mathcal{K}$ is $[\ell, u]$-sound and $[\ell, u]$-complete relative to $\mathcal{F}$ for saturated disjunctive statements. Now suppose that, for some finite set S, $\mathcal{F}_S$ does not have the Kronecker property on $\mathcal{S}^\ell(S)$. Hence, there does not exist a nonzero real number c_U and a set $D_U = \{d_{U,V} \in \mathbf{R} \mid V \notin \mathcal{S}^i(S)\}$ such that $F_{\mathcal{S}^\ell(S),U,c_U,D_U} \in \mathcal{F}$. For each V in $\mathcal{S}^\ell(S) - \{U\}$, we know that $U \neq V$ and, hence, $V \nsubseteq U$

or $U \nsubseteq V$ or both. We now construct a set of saturated disjunctive statements $\mathcal{C}$ such that $\mathcal{L}(\mathcal{C}) = \mathcal{S}^\ell(S) - \{U\}$. We distinguish two cases.

Case 1: $V \nsubseteq U$. Then, for every $V \in \mathcal{S}^\ell(S) - \{U\}$, add a *saturated* disjunctive statement to $\mathcal{C}$ of the form $V \to \mathcal{Y}$ with $\bigcup \mathcal{Y} = \overline{V}$ and $|\mathcal{Y}| = \ell$. This disjunctive statement exists, has order ℓ, and is non-trivial. Since $V \nsubseteq U$, $V \in \mathcal{L}(V, \mathcal{Y})$, but $U \notin \mathcal{L}(V, \mathcal{Y})$.

Case 2: $U \nsubseteq V$. Then, for every $V \in \mathcal{S}^\ell(S) - \{U\}$, add a *saturated* disjunctive statement to $\mathcal{C}$ of the form $V \to \{U - V\} \cup \mathcal{Y}$ with $(U - V) \cup \bigcup \mathcal{Y} = \overline{V}$ and $|\mathcal{Y}| = \ell - 1$. This disjunctive statement exists, has order ℓ, and is non-trivial. Since $U \nsubseteq V$, $V \in \mathcal{L}(V, \mathcal{Y}')$, but $U \notin \mathcal{L}(V, \mathcal{Y}')$.

Hence, $\mathcal{L}(\mathcal{C}) = \mathcal{S}^\ell(S) - \{U\}$. Now, consider the saturated disjunctive statement $U \to \mathcal{Z}$ with $\bigcup \mathcal{Z} = \overline{U}$ and $|\mathcal{Z}| = \ell$. This disjunctive statement exists, has order ℓ, and is non-trivial. Since $U \in \mathcal{L}(U, \mathcal{Z}')$, $\mathcal{L}(U, \mathcal{Z}) \nsubseteq \mathcal{L}(\mathcal{C})$. By Theorem 3.10, $\mathcal{C} \nvdash_{\mathcal{K}} U \to \mathcal{Z}$.

We now show that $\mathcal{C} \models_{\mathcal{F}} U \to \mathcal{Z}$ to obtain the desired contradiction. Thus, let F be in $\mathcal{F}_S$, and assume that F satisfies all disjunctive statements in $\mathcal{C}$. Since $\mathcal{K}$ is $[\ell, u]$-sound relative to $\mathcal{F}$ for saturated disjunctive statements, we know, by Theorem 3.13, that $\mathcal{F}$ has the $[\ell, u]$-zero-density property for saturated statements. Thus, $\Delta F(X) = 0$ for each $X \in \mathcal{S}^\ell(S)$ with $X \neq U$. But, then, $\Delta F(U) = 0$, since, otherwise, there would exist a nonzero real number $c_U = \Delta F(U) \neq 0$ and a set $D_U = \{d_{U,V} \in \mathbf{R} \mid V \notin \mathcal{S}^\ell(S)\}$ such that $F_{\mathcal{S}^\ell(S), U, c_U, D_U} = F \in \mathcal{F}$, a contradiction. Hence, F must be a function whose density is zero on every set in $\mathcal{S}^\ell(S)$. Thus, F satisfies $U \to \mathcal{Z}$, whence $\mathcal{C} \models_{\mathcal{F}} U \to \mathcal{Z}$.

Statement (1) is a special case of statement (2), except when $i = 1$. In this case, however, the construction of the set $\mathcal{C}$ in the proof of (2) also works for disjunctive statements that do not have to be saturated.

From Proposition 3.23, the following is easily shown.

Theorem 3.24. *Let $\ell, u \in \mathbf{N}$ with $2 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. If $\mathcal{K}$ is $[\ell, u]$-sound and $[\ell, u]$-complete relative to $\mathcal{F}$ for saturated disjunctive statements, then $\mathcal{K}$ is $[\ell, u]$-complete relative to $\mathcal{F}$.*

EXAMPLE 3.25. Inference system $\mathcal{G}$ is $[2, 2]$-complete for both the implication problem for multivalued dependencies and the implication problem for saturated conditional independence statements [**15**]. By Lemma 3.16, $\mathcal{K}$ is also $[2, 2]$-complete for these implication problems for saturated statements. By Example 3.15, $\mathcal{K}$ is also $[2, 2]$-sound for these implication problems for saturated statements. Using Theorem 3.24, we may thus conclude that $\mathcal{K}$ is $[2, 2]$-complete for both the implication problem for embedded multivalued dependencies and unrestricted conditional independence statements.

CHAPTER 4

Non-Existence of Finite Axiomatizations

We have presented properties of classes of real-valued functions that imply the soundness and completeness of inference systems for the implication problems for disjunctive statements. However, in some cases these properties will either not hold or be difficult to verify. There are classes of real-valued functions, for example, the class of multi-information functions, for which inference system $\mathcal{K}$ is complete but not sound. Before one tries to develop inference systems that are both sound and complete, one needs to consider the possibility that such a finite inference system may not exist. While this would be a "negative" result, much like proving the undecidability of a computational problem, it provides insights into the theoretical characteristics of implication problems and saves the researcher from a futile search for a non-existing solution.

There are several open problems concerning the existence of a finite complete axiomatization. On the one hand, the implication problem for measure-based constraints on the class of supermodular (submodular) functions (Sayrafi et al. [**40**]) and the implication problem for interaction statements relative to the class of all supermodular (submodular) worth functions were both not known to have a finite complete axiomatization. On the other hand, non-existence results are known for the implication problem for conditional independence statements (Studený [**44**]) and embedded multivalued dependencies (Sagiv [**35**]). We wish to develop a theory that captures these cases and solves some open problems in a general framework.

In this chapter, we will therefore specify properties of real-valued functions that imply the non-existence of a finite axiomatization for the implication problem for disjunctive statements interpreted as additive constraints on specific classes of real-valued functions. These properties hold for many classes of real-valued functions and will allow us to solve some open problems. It builds on and generalizes previous ideas of non-existence proofs in the areas of embedded multivalued dependencies [**35**] and conditional independence in the area of reasoning under uncertainty [**44**]. Again, the developed framework is applicable to implication problems that can be reduced to the implication problem on additive constraints.

1. Properties for Non-Axiomatizability of Implication Problems

In this section, we will specify properties that together imply the non-existence of a finite, complete axiomatization. We will also derive various propositions that will allow us to imply the non-existence of a finite axiomatization from properties introduced in previous chapters.

Let us first consider the following inference rule, Λ^n $(n \geq 3)$, which will play a central role in the development of our theory:

$$\begin{array}{lcl} A_1 & \rightarrow & \{A_2\} \cup \mathcal{Y} \\ A_2 & \rightarrow & \{A_3\} \cup \mathcal{Y} \\ & \vdots & \\ A_{n-1} & \rightarrow & \{A_n\} \cup \mathcal{Y} \\ A_n & \rightarrow & \{A_1\} \cup \mathcal{Y} \\ \hline A_1 & \rightarrow & \{A_n\} \cup \mathcal{Y} \end{array}$$

Definition 4.1 (Dual Kronecker). Let S be a finite set and let $\mathcal{F}_S$ be a set of real-valued functions over S. We say that $\mathcal{F}_S$ has the dual Kronecker property if, for every $U \subseteq S$ with $|U| \geq 2$, there exists a function $F^U \in \mathcal{F}_S$ and a real number $c \neq F^U(U)$ such that, for all V with $U \nsubseteq V \subseteq S$, $F^U(V) = c$.

The following theorem establishes properties that imply the non-existence of a finite axiomatization of the implication problem for disjunctive statements interpreted as additive constraints on specific classes of real-valued functions. In the remainder of the paper, we will show how these properties can be used in meaningful ways to derive new insights into open problems.

THEOREM 4.2. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. If all of the following statements hold, then there does not exist a finite, complete axiomatization for the implication problem for disjunctive statements of order at least ℓ and at most u relative to $\mathcal{F}$:*

1. *Inference rule Λ^n is $[\ell, u]$-sound relative to $\mathcal{F}$ for every $n \geq 3$;*
2. *$\mathcal{K}$ is $[\ell, u]$-complete relative to $\mathcal{F}$; and*
3. *for each finite set S, $\mathcal{F}_S$ has the dual Kronecker property.*

Proof: Assume to the contrary that there does exist a finite, complete axiomatization $\mathcal{X}$. Then, there must be $n \geq 3$ such that, for all $m > n$, there is no inference rule in $\mathcal{X}$ with m antecedents. Hence, for $m > n$, $\Lambda^m \notin \mathcal{X}$. Fix such m arbitrarily. Now, let S be a finite set with $|S| > m$ and $|S| \geq u$, and let $\mathcal{C} = \{\{a_1\} \rightarrow \{\{a_2\}\} \cup \mathcal{Y}, \ldots, \{a_{m-1}\} \rightarrow \{\{a_m\}\} \cup \mathcal{Y}, \{a_m\} \rightarrow \{\{a_1\}\} \cup \mathcal{Y}\}$, where $\mathcal{Y} \subseteq 2^S$ contains only singletons, $a_i \in S$ for all $i = 1, \ldots, m$, and $|\mathcal{Y}| = u - 1$. Since, by assumption, Λ^m is $[\ell, u]$-sound, we know that $\mathcal{C} \models_{\mathcal{F}} \{a_1\} \rightarrow \{\{a_m\}\} \cup \mathcal{Y}$.

We will now show that every proper subset of $\mathcal{C}$ does not imply *any* non-trivial disjunctive statement relative to $\mathcal{F}$ of order at least ℓ and at most u, using properties (2) and (3). Of course, this then implies that $\mathcal{C} \nvdash_{\mathcal{X}} a_1 \rightarrow \{a_m\} \cup \mathcal{Y}$, because $\mathcal{X}$ contains only inference rules with at most $m-1$ antecedents, contradicting the completeness of $\mathcal{X}$.

It is sufficient to show that every proper subset of $\mathcal{C}$ of size $m-1$ does not imply any non-trivial disjunctive statement of order at least ℓ and at most u. Hence, without loss of generality, let $\mathcal{C}' = \{\{a_1\} \rightarrow \{\{a_2\}\} \cup \mathcal{Y}, \ldots, \{a_{m-1}\} \rightarrow \{\{a_m\}\} \cup \mathcal{Y}\}$ be one of these proper subsets. Assume that $\mathcal{C}'$ implies the *novel* disjunctive statements $A \rightarrow \{B\} \cup \mathcal{Z}$.

If $|\mathcal{Z}| < u-1$, there exists an element U in $\mathcal{L}(A, \{B\} \cup \mathcal{Z})$ with $|U| = |S| - |\mathcal{Z}| - 1 > |S| - u$, by Definition 2.6. Since, also by Definition 2.6, the cardinality of the elements in $\mathcal{L}(\mathcal{C}')$ is at most $|S| - u$, we have, by Theorem 3.10 and by the $[\ell, u]$-completeness of $\mathcal{K}$, that $A \rightarrow \{B\} \cup \mathcal{Z}$ is *not* logically implied by $\mathcal{C}'$. Hence, $|\mathcal{Z}| = u - 1$. We distinguish two cases.

Case 1: At least one of the sets in $\{A, B\} \cup \mathcal{Z}$ has cardinality at least 2. Then, $|A \cup B \cup \bigcup \mathcal{Z}| > |\{a_i\} \cup \{a_{i+1}\} \cup \bigcup \mathcal{Y}|$, for all $i = 1, \ldots, m-1$ and, thus, $A \cup B \cup \bigcup \mathcal{Z} \nsubseteq$

$\{a_i\} \cup \{a_{i+1}\} \cup \bigcup \mathcal{Y}$, for all $i = 1, \ldots, m-1$. Now, one can verify that the dual Kronecker function $F^{A \cup B \cup \bigcup \mathcal{Z}}$ satisfies all the elements in $\mathcal{C}'$, but does not satisfy $A \rightarrow \{B\} \cup \mathcal{Z}$. Hence, $A \rightarrow \{B\} \cup \mathcal{Z}$ is not implied by $\mathcal{C}'$, contradicting the assumption.

Case 2: A, B, and all elements of $\mathcal{Z}$ have cardinality 1. Since $\mathcal{K}$ is $[\ell, u]$-complete, we know that $A = \{a_i\}$ for some i, $1 \leq i \leq n-1$. Similarly, $\{B\} \cup \mathcal{Z} = \{\{a_j\}\} \cup \mathcal{Y}$ for some j, $2 \leq j \leq n$, since, otherwise, $S - (A \cup \bigcup \mathcal{Y}) \in \mathcal{L}(A, \{B\} \cup \mathcal{Z})$, but $S - (A \cup \bigcup \mathcal{Y}) \notin \mathcal{L}(\mathcal{C}')$ which, by Definition 2.6, would contradict the $[\ell, u]$-completeness of $\mathcal{K}$. Furthermore, $B \neq \{a_{i+1}\}$, since $A \rightarrow \{B\} \cup \mathcal{Z}$ would then be an element of $\mathcal{C}'$. Hence, $A \cup B \cup \bigcup \mathcal{Z} \nsubseteq \{a_i\} \cup \{a_{i+1}\} \cup \bigcup \mathcal{Y}$, for all $i = 1, \ldots, m-1$. Now, one can verify that the dual Kronecker function $F^{A \cup B \cup \bigcup \mathcal{Z}}$ satisfies all the elements in $\mathcal{C}'$, but does not satisfy $A \rightarrow \{B\} \cup \mathcal{Z}$. Hence, $A \rightarrow \{B\} \cup \mathcal{Z}$ is not implied by $\mathcal{C}'$, contradicting the assumption.

So, no proper subset of $\mathcal{C}$ implies a non-trivial disjunctive statement relative to $\mathcal{F}$ of order at least ℓ and at most u, which completes the proof.

We will now show that inference rule Λ^n, $n \geq 3$, is sound for an important class of real-valued functions, namely Choquet capacities, which we have introduced in Chapter 2. Since many classes of real-valued functions consist of Choquet capacities, the proposition will allow us to infer the soundness of inference rule Λ^n for these classes of real-valued functions.

PROPOSITION 4.3. *Let k be a number, $k \geq 1$, and let $\mathcal{F}$ be a class of real-valued functions. If, for each finite set S, and for each real-valued function F in $\mathcal{F}_S$, F is a positive (negative) k-alternating capacity, then Λ^n is $[k,k]$-sound relative to $\mathcal{F}$, for each $n \geq 3$.*

Proof: Let S be a finite set. We begin by proving that if for each real-valued function F in $\mathcal{F}_S$, F is a positive k-alternating capacity, then Λ^n is $[k,k]$-sound relative to $\mathcal{F}$, for each $n \geq 3$.

Let F be a positive k-alternating capacity that satisfies all the antecedents of the inference rule Λ^n. By Definition 2.2 and Definition 2.14 we have that

$$
\begin{gathered}
\Delta^{\mathcal{Y}} F(A_1) - \Delta^{\mathcal{Y}} F(A_1 \cup A_2) = 0; \\
\Delta^{\mathcal{Y}} F(A_2) - \Delta^{\mathcal{Y}} F(A_2 \cup A_3) = 0; \\
\vdots \\
\Delta^{\mathcal{Y}} F(A_{n-1}) - \Delta^{\mathcal{Y}} F(A_{n-1} \cup A_n) = 0; \text{ and} \\
\Delta^{\mathcal{Y}} F(A_n) - \Delta^{\mathcal{Y}} F(A_n \cup A_1) = 0.
\end{gathered}
$$

Since each of the equations is equal to zero, the sum of all equations is also equal to zero. Consequently, by a simple rearrangement of the summands we also have that

$$
\begin{gathered}
\Delta^{\mathcal{Y}} F(A_1) - \Delta^{\mathcal{Y}} F(A_1 \cup A_2) + \\
\Delta^{\mathcal{Y}} F(A_2) - \Delta^{\mathcal{Y}} F(A_2 \cup A_3) + \\
\vdots \\
\Delta^{\mathcal{Y}} F(A_{n-1}) - \Delta^{\mathcal{Y}} F(A_{n-1} \cup A_n) + \\
\Delta^{\mathcal{Y}} F(A_n) - \Delta^{\mathcal{Y}} F(A_n \cup A_1) \\
= \\
\Delta^{\mathcal{Y}} F(A_1) - \Delta^{\mathcal{Y}} F(A_1 \cup A_n) + \\
\Delta^{\mathcal{Y}} F(A_2) - \Delta^{\mathcal{Y}} F(A_2 \cup A_1) + \\
\vdots \\
\Delta^{\mathcal{Y}} F(A_{n-1}) - \Delta^{\mathcal{Y}} F(A_{n-1} \cup A_{n-2}) + \\
\Delta^{\mathcal{Y}} F(A_n) - \Delta^{\mathcal{Y}} F(A_n \cup A_{n-1}) \\
= \\
0.
\end{gathered}
$$

By assumption F is a k-alternating positive capacity, and we have that $\Delta^{\{Y\}\cup\mathcal{Y}}F(X) \geq 0$ for all $X, Y \subseteq S$. Hence, it follows that

$$\begin{gathered}
\Delta^{\mathcal{Y}}F(A_1) - \Delta^{\mathcal{Y}}F(A_1 \cup A_n) = 0; \\
\Delta^{\mathcal{Y}}F(A_2) - \Delta^{\mathcal{Y}}F(A_2 \cup A_1) = 0; \\
\vdots \\
\Delta^{\mathcal{Y}}F(A_{n-1}) - \Delta^{\mathcal{Y}}F(A_{n-1} \cup A_{n-2}) = 0; \text{ and} \\
\Delta^{\mathcal{Y}}F(A_n) - \Delta^{\mathcal{Y}}F(A_n \cup A_{n-1}) = 0.
\end{gathered}$$

Therefore, F also satisfies the disjunctive statement $A_1 \to \{A_n\} \cup \mathcal{Y}$. This concludes the proof. The proof for negative Choquet capacities is analogous and omitted.

Based on results in the previous chapter, we can state several propositions that allow us to derive non-existence proofs for the finite axiomatizability of implication problems on disjunctive statements. These results are based on Theorem 4.2, Theorem 3.24, and Proposition 4.3.

PROPOSITION 4.4. *Let $\ell, u \in \mathbf{N}$ with $1 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. If all of the following statements hold, then there does not exist a finite, complete axiomatization for the implication problem for additive constraints on $\mathcal{F}$ for disjunctive statements of order at least ℓ and at most u:*

1. *Every $F \in \mathcal{F}$ is a (negative) positive k-alternating capacity, for $\ell \leq k \leq u$;*
2. *For each finite set S, $\mathcal{F}_S$ has the Kronecker property on $\mathcal{S}^{\ell}(S)$; and*
3. *For each finite set S, $\mathcal{F}_S$ has the dual Kronecker property.*

Proof: Since, by assumption, each $F \in \mathcal{F}$ is a (negative) positive k-alternating capacity, for $\ell \leq k \leq u$, we have that Λ^n is $[k, k]$-sound relative to $\mathcal{F}$, for each $n \geq 3$, by Proposition 4.3. Since, by assumption, $\mathcal{F}_S$ has the Kronecker property on $\mathcal{S}^{\ell}(S)$, we have that $\mathcal{K}$ is $[k, k]$-complete relative to $\mathcal{F}$, by Theorem 3.20. Now, by Theorem 4.2, the statement of the proposition follows.

PROPOSITION 4.5. *Let $\ell, u \in \mathbf{N}$ with $2 \leq \ell \leq u$ be bounds on the order of the disjunctive statements, and let $\mathcal{F}$ be a class of real-valued functions. If all of the following statements hold, then there does not exist a finite, complete axiomatization for the implication problem for additive constraints on $\mathcal{F}$ for disjunctive statements of order at least ℓ and at most u:*

(1) *Every $F \in \mathcal{F}$ is a (negative) positive k-alternating capacity, for $\ell \leq k \leq u$;*
(2) *$\mathcal{K}$ is $[\ell, u]$-sound and $[\ell, u]$-complete relative to $\mathcal{F}$ for saturated disjunctive statements; and*
(3) *For each finite set S, $\mathcal{F}_S$ has the dual Kronecker property.*

Proof: Since, by assumption, each $F \in \mathcal{F}$ is a (negative) positive k-alternating capacity, for $\ell \leq k \leq u$, we have that Λ^n is $[k, k]$-sound relative to $\mathcal{F}$, for each $n \geq 3$, by Proposition 4.3. Since, by assumption, $\mathcal{K}$ is $[\ell, u]$-sound and $[\ell, u]$-complete relative to $\mathcal{F}$ for saturated disjunctive statements, we have that $\mathcal{K}$ is $[k, k]$-complete relative to $\mathcal{F}$ by Theorem 3.24. Now, by Theorem 4.2, the statement of the proposition follows.

2. Application of the Non-Axiomatizability Results

We conclude this chapter with some examples of bounded implication problems for which no finite, complete axiomatization exists.

The first one relates to **conditional independence**. Let $\mathcal{M}$ be the class of multi-information functions induced by the class of discrete probability measures. We know that $\mathcal{K}$ is $[2, 2]$-complete relative to $\mathcal{M}$ [**30**]. We also know that, for each finite set S, $\mathcal{M}_S$ has the dual Kronecker property [**46**]. Finally, since every $M \in \mathcal{M}$ is supermodular (i.e., a positive 2-alternating capacity), we know that inference rule Λ^n is $[2, 2]$-sound relative to $\mathcal{M}$. Hence, as was previously shown [**44**], there exists no finite, complete axiomatization for the implication problem for additive constraints on $\mathcal{M}$ of order exactly 2. All these conditions do also hold for the class of multi-information functions induced by the class of binary discrete probability measures. Hence, there exists no finite, complete axiomatization for the implication problem for CI statements relative to the class of binary discrete probability measures, which solves an open problem [**15**].

We next turn to **Choquet capacities**. Let $k \geq 1$ be a natural number, and let $\mathcal{F}$ be the class of all positive (negative) k-alternating capacities. We know that, for each finite set S, $\mathcal{F}_S$ has the Kronecker property on $\mathcal{S}^k(S)$. We know that $\mathcal{F}_S$ has the dual Kronecker property. From Proposition 4.3, it finally follows that Λ^n is $[k,k]$-sound relative to $\mathcal{F}$, for each $n \geq 3$. Hence, there exists no finite, complete axiomatization for the implication problem for additive constraints on $\mathcal{F}$ of order exactly k. In particular, there exists no finite, complete axiomatization of the implication problem for additive constraints of order exactly 1 for the class of increasing functions, and of order 2 for the class of all supermodular functions, respectively. This answers the open question [**40**] whether there exists a finite, complete axiomatization for measure-based constraints on the class of supermodular (submodular) functions in the negative.

Finally, we revisit **cooperative game theory**. It follows from the previous example that the implication problem for interaction statements relative to the class of all supermodular (submodular) worth functions has no finite complete axiomatization.

CHAPTER 5

Conditional Independence and Reasoning under Uncertainty

Conditional independence is an important concept in many calculi for dealing with knowledge and uncertainty in artificial intelligence. The notion plays a fundamental role for learning and reasoning in probabilistic systems which are successfully employed in areas such as computer vision, computational biology, and robotics. Hence, new theoretical findings and algorithmic improvements have the potential to impact many fields of research. A central issue for reasoning about conditional independence is the *probabilistic conditional independence implication problem*, that is, to decide whether a CI statement is entailed by a set of other CI statements relative to the class of discrete probability measures. While it remains open whether this problem is decidable, it is known that there exists no finite, sound and complete inference system (Studený [**44**]). However, there exist finite sound inference systems that have attracted special interest. The most prominent is the *semi-graphoid* axiom system (see Figure 5.1), which was introduced as a set of sound inference rules relative to the class of discrete probability measures (Pearl [**31**]). One of the main contributions of this work is to extend the semi-graphoids to a *finite* inference system, denoted by $\mathcal{A}$, which we will show to be (1) sound and complete for saturated CI statements, (2) complete for general CI statements, and (3) sound and complete for stable CI statements, a notion that has been intruduced by de Waal and van der Gaag [**9**].

The techniques we use to obtain these results are based on the theoretical framework we have introduced in earlier chapters. By interpreting conditional independence statements as disjunctive statements and associating semi-lattices with these statements, $\mathcal{A}$ is shown to be sound and complete relative to certain inclusion relationships on these semi-lattices. To make the connection between the theoretical framework for disjunctive statements and the conditional independence implication problem, we introduce the concept of

$I(A, \emptyset \mid C)$	**Triviality**
$I(A, B \mid C) \rightarrow I(B, A \mid C)$	**Symmetry**
$I(A, B \cup D \mid C) \rightarrow I(A, D \mid C)$	**Decomposition**
$I(A, B \mid C \cup D)$ & $I(A, D \mid C) \rightarrow I(A, B \cup D \mid C)$	**Contraction**
$I(A, B \cup D \mid C) \rightarrow I(A, B \mid C \cup D)$	**Weak union**

FIGURE 5.1. The semi-graphoid axiom system, is sound, but not complete, for the implication problem for unrestricted CI statements.

multi-information functions induced by probability measures (Studený [**46**]). This class of real-valued functions allows us to link the implication problem for additive constraints on real-valued functions to the probabilistic CI implication problem. The combination of the lattice-inclusion techniques and the completeness result for conditional independence statements allows us to derive criteria that can be used to falsify instances of the implication problem. Furthermore, we introduce a novel validation algorithm that leverages the falsification algorithm and represents implication problems as instances of linear programming problems. We show experimentally that the falsification and validation criteria, some of which can be tested in polynomial time, work very effectively, and, where possible, we relate the experimental results to those obtained from a *racing algorithm* introduced by Bouckaert and Studený [**3**].

1. Conditional Independence Statements: Inference System $\mathcal{A}$

We define CI statements and introduce the finite inference system $\mathcal{A}$ for reasoning about the conditional independence implication problem. Note that inference system $\mathcal{A}$ is based on inference system $\mathcal{K}$, where the order of the disjunctive statements is restricted to be exactly 2. We will often write AB for the union $A \cup B$, ab for the set $\{a, b\}$, and a for the singleton set $\{a\}$ whenever the interpretation is clear from the context. Throughout

$I(A, \emptyset \mid C)$	**Triviality**
$I(A, B \mid C) \rightarrow I(B, A \mid C)$	**Symmetry**
$I(A, BD \mid C) \rightarrow I(A, D \mid C)$	**Decomposition**
$I(A, B \mid CD) \wedge I(A, D \mid C) \rightarrow I(A, BD \mid C)$	**Contraction**
$I(A, B \mid C) \rightarrow I(A, B \mid CD)$	**Strong union**
$I(A, B \mid C) \wedge I(A, D \mid C) \rightarrow I(A, BD \mid C)$	**Composition**
$I(A, B \mid C) \wedge I(D, E \mid AC) \wedge I(D, E \mid BC) \rightarrow I(D, E \mid C)$	**Strong contraction**

FIGURE 5.2. The inference rules of system $\mathcal{A}$.

the chapter, S will denote a finite implicit set of statistical variables. Let us first recall the definition of a conditional independence statement.

DEFINITION 5.1. The expression $I(A, B|C)$, with A, B, and C pairwise disjoint subsets of S, is called a *conditional independence (CI) statement*. If $ABC = S$, we say that $I(A, B|C)$ is *saturated*. If either $A = \emptyset$ and/or $B = \emptyset$, we say that $I(A, B|C)$ is *trivial*.

Once again, note that (saturated) conditional independence statements can be seen as (saturated) disjunctive statements.

The set of inference rules in Figure 5.2 will be denoted by $\mathcal{A}$. Again, note that it is the set of inference rules of inference system $\mathcal{K}$ specialized to the $[2, 2]$-bounded implication problem (see Figure 3.2), with the additional inference rule **symmetry** which is implicitly sound for disjunctive statements. Furthermore, note that we adopt terminology that is most commonly used in the artificial intelligence community. More specifically, the names of some of the inference rules differ. For instance, what was referred to as **augmentation** in previous chapters is now called **strong union**. The **triviality**, **symmetry**, **decomposition**, and **contraction** rules are part of the semi-graphoid axioms (Pearl [**31**]). Note that **strong union** is not a sound inference rule relative to the class of discrete probability measures. The derivability of a CI statement c from a set of CI statements $\mathcal{C}$ under the inference rules of system $\mathcal{A}$ is denoted by $\mathcal{C} \vdash_{\mathcal{A}} c$. The *closure* of $\mathcal{C}$ under $\mathcal{A}$, denoted $\mathcal{C}^+$, is the set $\{c \mid \mathcal{C} \vdash_{\mathcal{A}} c\}$.

2. The Conditional Independence Implication Problem

While the theoretical framework presented so far has been concerned with the implication problem for additive constraints, it also captures the implication problem for conditional independence statements. The link between these two problems is made possible with the concept of *multi-information functions* (Studený [**46**]) induced by probability measures. In this chapter, we will restrict our discussion to the class of *discrete* probability measures. Let us begin by introducing the notion of a probability model.

Definition 5.2. A *probability model* over $S = \{s_1, \ldots, s_n\}$ is a pair (dom, P), where dom is a domain mapping that maps each s_i to a finite domain $dom(s_i)$, and P is a probability measure having $dom(s_1) \times \cdots \times dom(s_n)$ as its sample space. For $A = \{a_1, \ldots, a_k\} \subseteq S$, we will say that $\mathbf{a}$ is a domain vector of A if $\mathbf{a} \in dom(a_1) \times \cdots \times dom(a_k)$.

In what follows, we will only refer to probability measures, keeping their probability models implicit.

Definition 5.3. Let $I(A, B|C)$ be a CI statement, and let P be a probability measure. We say that P *satisfies* $I(A, B|C)$, and write $\models_P I(A, B|C)$, if for every domain vector $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ of A, B, and C, respectively, $P(\mathbf{c})P(\mathbf{a}, \mathbf{b}, \mathbf{c}) = P(\mathbf{a}, \mathbf{c})P(\mathbf{b}, \mathbf{c})$.

Relative to the notion of *satisfaction* we can now define the *probabilistic conditional independence implication problem.*

Definition 5.4 (Probabilistic conditional independence implication problem). Let $\mathcal{C}$ be a set of CI statements and let c be a CI statement. We say that $\mathcal{C}$ *implies* c relative to the class of discrete probability measures, and write $\mathcal{C} \models c$, if every discrete probability measure that *satisfies* the CI statements in $\mathcal{C}$ also *satisfies* the CI statement c.

DEFINITION 5.5. The *conditional independence (CI) implication problem* is the problem of deciding the language

$$\{(S, \mathcal{C}, c) \mid \mathcal{C} \text{ a set of CI statements over } S,\ c \text{ a CI statement over } S,\ \mathcal{C} \models c\}.$$

Next, we define the *multi-information function* induced by a probability measure (Studený [**46**]), which is based on the Kullback-Leibler divergence (Kullback and Leibler [**20**]).

DEFINITION 5.6. Let P and Q be two probability measures over a discrete sample space, and let P be absolutely continuous with respect to Q. Then, the relative entropy (Kullback-Leibler divergence) H is defined as

$$H(P|Q) := \sum_{\mathbf{x}} \{P(\mathbf{x}) \log \frac{P(\mathbf{x})}{Q(\mathbf{x})},\ P(\mathbf{x}) > 0\},$$

with $\mathbf{x}$ ranging over all elements of the discrete sample space.

DEFINITION 5.7. Let P be a probability measure, and let H be the relative entropy. The *multi-information function* $M_P : 2^S \to [0, \infty]$ induced by P is defined as

$$M_P(A) := H(P^A | \prod_{a \in A} P^{\{a\}}),$$

for each non-empty subset A of S and $M_P(\emptyset) = 0$.[1]

The class of multi-information functions induced by the class of discrete probability measures will be denoted by $\mathcal{M}$. We can now state the a result of Studený that couples the probabilistic CI implication problem with the implication problem for additive constraints relative to $\mathcal{M}$.

THEOREM 5.8 (Studený [**46**]). *Let $\mathcal{C}$ be a set of CI statements and let c be a CI statement. Then, $\mathcal{C} \models_{\mathcal{M}} c$ if and only if $\mathcal{C} \models c$.*

[1]Here, P^A and $P^{\{a\}}$ denote the marginal probability measures of P over A and $\{a\}$, respectively.

3. Saturated Conditional Independence Statements: Soundness and Completeness of inference system $\mathcal{A}$

In the previous section, we established the link between the theoretical framework for disjunctive statements and the probabilistic conditional independence implication problem. In this section, we will use this connection to show that inference system $\mathcal{A}$ is sound and complete for the probabilistic CI implication problem for *saturated* CI statements. We recall that a CI statement $I(A, B|C)$ is saturated if $ABC = S$. We begin by showing the following technical lemma.

LEMMA 5.9. *The class of multi-information functions $\mathcal{M}$ induced by the class of discrete probability measures has the zero-density property with respect to saturated CI statements.*

Proof: We have to show that, for each saturated CI statements c, for each $M \in \mathcal{M}$, and for each $U \in \mathcal{L}(c)$, if M satisfies c, then $\Delta M(U) = 0$. The semi-graphoid inference rules are sound relative to the class of probability measures. Hence, in particular, by Theorem 5.8, *weak union* is sound relative to $\mathcal{M}$, i.e., $\{I(AD, B|C)\} \models_{\mathcal{M}} I(A, B|CD)$. Let $M \in \mathcal{M}$, let ΔM be the corresponding density function, and let M satisfy the saturated CI statement $I(A, B|C)$ with $ABC = S$. In addition, let $I(A, B|C)$ be non-trivial since the proposition is obviously true for trivial CI statements. We will prove by downward induction on the semi-lattice $\mathcal{L}(A, B|C)$ that $\Delta M(U) = 0$ for each $U \in \mathcal{L}(A, B|C)$. Note that this proof is similar to the proof of Theorem 3.14.

For the base case, we show for each $W \in \mathcal{W}(A, B|C)$, that $\Delta M(\overline{W}) = 0$. Let $W = \{a, b\}$. By repeatedly applying *weak union* we can derive $\models_M I(a, b|\overline{W})$ because $ABC = S$. Now, since $\mathcal{L}(a, b|\overline{W}) = \{\overline{W}\}$, we can conclude that $\Delta M(\overline{W}) = 0$.

For the induction step, let $V \in \mathcal{L}(A, B|C)$. The induction hypothesis states that $\Delta M(U) = 0$ for each $U \in \mathcal{L}(A, B|C)$ with U a strict superset of V. From the given CI statement $I(A, B|C)$, we can derive, again by *weak union*, $I(A', B'|V)$ with $VA'B' = S$, $A \subseteq A'$, and $B \subseteq B'$ since $V - C \subseteq AB$. Since $\mathcal{L}(A', B'|V)$ contains only V and strict supersets V'

of V, with $V' \in \mathcal{L}(A, B|C)$, we can conclude that $\sum_{U \in \mathcal{L}(A', B')V} \Delta F(U) = \Delta F(V) = 0$, by the induction hypothesis.

We are now in the position to prove that inference system $\mathcal{A}$ is sound and complete for the probabilistic implication problem for *saturated* conditional independence statements.

THEOREM 5.10. *Inference system $\mathcal{A}$ is sound and complete for the probabilistic conditional independence implication problem for saturated CI statements.*

Proof: The soundness follows directly from Lemma 5.9, Theorem 3.13, and Theorem 5.8. To show completeness, notice that the semi-graphoid axioms are derivable under inference system $\mathcal{A}$. Furthermore, Geiger and Pearl proved that the semi-graphoid axioms are complete for the probabilistic conditional independence implication problem for *saturated* CI statements (Geiger and Pearl [**15**]). This concludes the proof.

We will now show that inference system $\mathcal{A}$ is complete for the probabilistic conditional independence implication problem. We first prove that, for any finite set S, the class of multi-information functions $\mathcal{M}$ induced by discrete probability measures over S has the Kronecker property on $\mathcal{S}^i(S)$. The completeness of inference system $\mathcal{A}$ for the CI implication problem can now be proved based on Theorem 3.24 because we have shown that $\mathcal{A}$ is sound and complete with respect to saturated conditional independence statements.

THEOREM 5.11. *Inference system $\mathcal{A}$ is complete for the probabilistic conditional independence implication problem.*

Proof: We know from Theorem 5.10 that inference system $\mathcal{A}$ is sound and complete for the probabilistic CI implication problem for saturated CI statements. Now, by Theorem 3.24, the statement follows.

EXAMPLE 5.12. (Studený [**46**]) described the following sound inference rule relative to discrete probability measures which refuted the conjecture (Pearl [**31**]) that the semi-graphoid axioms are complete for the probabilistic CI implication problem:

$$I(A,B|CD) \wedge I(C,D|A) \wedge I(C,D|B) \wedge I(A,B|\emptyset)$$
$$\rightarrow$$
$$I(C,D|AB) \wedge I(A,B|C) \wedge I(A,B|D) \wedge I(C,D|\emptyset).$$

By applying *strong contraction* to the statements $I(A,B|\emptyset), I(C,D|A)$, and $I(C,D|B)$, we can derive the statement $I(C,D|\emptyset)$. All the other statements can be derived using *strong union*.

Remark 5.13. The inference system $\mathcal{A}$ without *strong contraction* is *not* complete. The consequence $I(C,D|\emptyset)$ of the clause from Example 5.12 cannot be derived from the antecedents without *strong contraction*.

4. Complete Axiomatization of Stable Independence

When new information is available to a probabilistic system, the set of associated relevant CI statements changes dynamically. However, some of the CI statements will continue to hold. These CI statements were termed *stable* by de Waal and van der Gaag [**9**]. A first investigation of their structural properties was undertaken by Matúš who used the term *ascending* conditional independence (Matúš [**27**]). Every set of CI statements can be partitioned into its *stable* and *unstable* part. We will show that inference system $\mathcal{A}$ is sound and complete for the probabilistic CI implication problem for *stable* conditional independence statements.

Definition 5.14. Let $\mathcal{C}$ be a set of CI statements, and let $\mathcal{C}^{SG+}$ be the semi-graphoid closure of $\mathcal{C}$. Then $I(A,B|C)$ is said to be *stable* in $\mathcal{C}$, if $I(A,B|C') \in \mathcal{C}^{SG+}$ for all sets C' with $C \subseteq C' \subseteq S$.

Theorem 5.15. *Let $\mathcal{C}_S$ be a set of stable CI statements. Then, $\mathcal{A}$ is sound and complete for the probabilistic conditional independence implication problem for $\mathcal{C}_S$.*

Proof: The soundness follows from Theorem 3.13 and from *strong union* and *decomposition* being sound inference rules relative to $\mathcal{M}$ for stable CI statements. The completeness follows from Theorem 5.11.

The previous result is also interesting with respect to the problem of finding a minimal, non-redundant representation of stable independence relations. Here, lattice-inclusion can aid the lossless compaction of representations of stable CI statements: $\mathcal{L}(\mathcal{C}_S - \{c\}) = \mathcal{L}(\mathcal{C}_S)$ if and only if c is redundant in $\mathcal{C}_S$. In Chapter 6 we will return to the concept of stable independence and show how the completeness of inference system $\mathcal{A}$ and the semi-lattice characterization can be leveraged to create more concise representations of stable conditional independence structures, and to gain a deeper understanding of the concept.

5. Falsification Algorithm

Theorem 3.10 and Theorem 5.11 lend themselves to a *falsification algorithm*, that is, an algorithm which can falsify instances of the probabilistic conditional independence implication problem. For instance, the falsification algorithm can be used as a pre-processing step for a racing algorithm as proposed by Bouckaert and Studený [**3**]. We first consider the following corollary, which directly follows from these two results.

Corollary 5.16. *Let S be a finite set, let $\mathcal{C}$ be a set of CI statements over S, and let c be single CI statement over S. If $\mathcal{L}(\mathcal{C}) \not\supseteq \mathcal{L}(c)$, then $\mathcal{C} \not\models c$.*

However, if the the number of instances of the implication problem that can be falsified with the algorithm were, on average, only a small fraction of all those that are possibly falsifiable, the result would be disappointing from a practical point of view. Fortunately, we will not only be able to show that a large number of implications can be falsified by the "lattice-exclusion" criterion identified in Corollary 5.16, but also that polynomial time heuristics exist that provide good approximations of said criterion.

Falsification Criterion. `Input`: A set of CI statements $\mathcal{C}$ and a CI statement c. `Test`: if $\mathcal{L}(\mathcal{C}) \not\supseteq \mathcal{L}(c)$, return "false", else return "unknown."

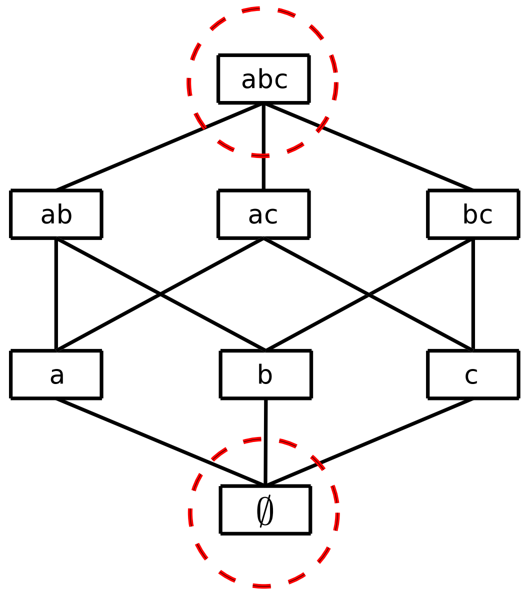

FIGURE 5.3. Hasse diagram of the lattice $[\emptyset, \{a,b,c\}]$. The circled elements are the meet (greatest lower bound) $\emptyset$ and the join (least upper bound) $\{a,b,c\}$ of the lattice. Semi-lattices of conditional independence statements always have a unique meet and one or more joins.

Heuristic 1. `Input`: A set of CI statements $\mathcal{C}$ and a CI statement $I(A,B|C)$. `Test`: if, for each $I(A',B'|C') \in \mathcal{C}$, we have that $C \not\supseteq C'$, return "false", else return "unknown."

Heuristic 2. `Input`: A set of CI statements $\mathcal{C}$, and a CI statement $I(A,B|C)$. `Test`: if there exists one $W \in \mathcal{W}(A,B|C)$ such that, for all $I(A',B'|C') \in \mathcal{C}$, we have that $W \notin \mathcal{W}(A',B'|C')$, return "false", else return "unknown."

It follows from Proposition 3.5 that if one of the two heuristics returns "false," then $\mathcal{L}(\mathcal{C}) \not\supseteq \mathcal{L}(c)$, and therefore $\mathcal{C} \not\models c$ by Corollary 5.16.

EXAMPLE 5.17. Let S be a finite set, and A, B, C, and D be pairwise disjoint subsets of S. The inference rule *intersection*,

$$I(A,B|DC) \wedge I(A,D|BC) \rightarrow I(A,BD|C),$$

is *not* sound relative to the class of discrete probability measures. Heuristic 1 can reject this instance of the implication problem in polynomial time in the size of S.

Remark 5.18. The falsification criterion leads in fact to a *family* of polynomial time heuristics. While Heuristic 1 checks if the unique *meet* (greatest lower bound) of the semi-lattice $\mathcal{L}(c)$ is not in $\mathcal{L}(\mathcal{C})$ and Heuristic 2 if the (potentially multiple) *joins* (least upper bounds) of the semi-lattice $\mathcal{L}(c)$ are not in $\mathcal{L}(\mathcal{C})$ (see Figure 5.3), we may select additional elements in the semi-lattice $\mathcal{L}(c)$ that are located between these two extrema to derive more falsification heuristics.

6. Validation Algorithm

In general, a validation algorithm takes as input an instance of the implication problem, consisting of a set of CI statements $\mathcal{C}$ and a single CI statement c over a finite set S, and accepts only if $\mathcal{C} \models c$. Of course, the algorithm not accepting an instance of the implication problem does not imply that the instance is invalid. Please note that one of the most prominent validation algorithms is the algorithm that computes the closure of the semi-graphoid axioms (Dawid [**8**], Pearl [**31**]). The semi-graphoid axioms are listed in Figure 5.1. However, the closure of the semi-graphoid axioms can only validate a small fraction of the set of verifiable instances. For instance, consider Example 5.12. The antecedents in this example cannot be derived by the semi-graphoid axioms even though they are implied. Before we derive our validation algorithm, we need some definitions of important technical concepts. We start by recalling the definition of the Möbius inversion of a real-valued function.

Definition 5.19. Let S be a finite set, and let F be a real-valued function over S. The *Möbius inversion* of F is the real-valued function ΔF defined by

$$\Delta F(X) = \sum_{X \subseteq U \subseteq S} (-1)^{|U|-|X|} F(U),$$

for each $X \subseteq S$.

Now, we have the following crucial relationship between a multi-information function, its Möbius inversion, and the semi-lattice of a CI statement.

LEMMA 5.20. *Let S be a finite set of random variables, let P be a discrete probability measure over S, let M_P be the multi-information function induced by P, let ΔM_P be the Möbius inversion of M_P, and let $I(A,B|C)$ be a CI statement over S. Then, the following statements are equivalent*

(1) *P satisfies $I(A,B|C)$;*
(2) *$M_P(ABC) + M_P(C) - M_P(AC) - M_P(BC) = 0$; and*
(3) $\displaystyle\sum_{U \in \mathcal{L}(C,\{A,B\})} \Delta M_P(U) = 0.$

Proof: Studený showed that (1) if and only if (2) [**46**]. In addition, we have that (2) if and only if (3), because $F(ABC) + F(C) - F(AC) - F(BC) = \sum_{U \in \mathcal{L}(C,\{A,B\})} \Delta F(U)$ for any real-valued function F. We refer the reader to Sayrafi and Van Gucht [**38**] for a proof of the latter statement.

We will now be able to harness the equivalences stated in the previous lemma to represent each set of CI statements $\mathcal{C}$ as a *minimal* sparse 0-1 matrix $\mathbf{A}$. Each instance of the implication problem with $\mathcal{C}$ as the set of antecedents will then correspond to a linear program with equality constraints specified by $\mathbf{A}$. Before we explain the construction of the constraint matrix $\mathbf{A}$, however, we have to define some additional technical concepts. For some of the following results, we need the concept of elementary CI statements, which are simply CI statements $I(a,b|K)$ with $a,b \in S$ and $K \subseteq S \setminus \{a,b\}$. We will write $\mathcal{B}(S)$ to denote the set of elementary CI statements over a finite set S.

DEFINITION 5.21. Let S be a finite set and let $\mathcal{C}$ be a set of CI statements over S. The set of *relevant* elementary CI statements $\mathcal{R}(\mathcal{C})$ is defined as follows:

$$\mathcal{R}(\mathcal{C}) = \{I(a,b|K) \in \mathcal{B}(S) \mid \mathcal{L}(K,\{\{a\},\{b\}\}) \subseteq \mathcal{L}(\mathcal{C})\}.$$

We will use the elementary CI statements in $\mathcal{R}(\mathcal{C})$ to construct the constraint matrix $\mathbf{A}$. Please note that $\mathcal{R}(\mathcal{C})$ is the set of elementary CI statements over S that remain (i.e., could not be falsified) after the application of the falsification algorithm that we introduced in this chapter. Hence, it follows that polynomial-time heuristics and SAT solvers can be used to compute the set $\mathcal{R}(\mathcal{C})$ efficiently for up to several hundreds of variables (Niepert and Van Gucht [**29**]). By Proposition 5.16, only CI statements $I(A, B|C)$ with $\mathcal{L}(C, \{A, B\}) \subseteq \mathcal{L}(\mathcal{C})$ can possibly be implied by $\mathcal{C}$. Now, using the concept of a semi-lattice, each of these candidate CI statements $c = I(A, B|C)$ can be written as a vector $\mathbf{v}_c$ relative to the space $\{0, 1\}^{\mathcal{L}(\mathcal{C})}$ as follows: for every $U \in \mathcal{L}(\mathcal{C})$ we have $\mathbf{v}_c(U) = 1$ if $U \in \mathcal{L}(C, \{A, B\})$ and $\mathbf{v}_c(U) = 0$ otherwise. The vector representation of a set of CI statements $\mathcal{C}$ can then be defined as the sum of the vectors corresponding to individual elements in $\mathcal{C}$: $\mathbf{v}_{\mathcal{C}} = \sum_{c \in \mathcal{C}} \mathbf{v}_c$. This is equivalent to the definition of an imset (Studený [**46**]), except that we use the Möbius inversion to avoid negative elements in the vector representation and that the vector representation is constructed relative to the union of semi-lattices $\mathcal{L}(\mathcal{C})$ of the CI statements in $\mathcal{C}$. Given these definitions of vector representations for individual CI statements and for sets of CI statements, we can state the following crucial result.

PROPOSITION 5.22. *Let S be a finite set, let $\mathcal{C}$ be a set of CI statements, let c be a single CI statement over S, and let $\mathbb{Q}^+$ be the non-negative rational numbers. Then, $\mathcal{C} \models c$ if*

$$\mathbf{v}_{\mathcal{C}} = \mathbf{v}_c + \textstyle\sum_{r \in \mathcal{R}(\mathcal{C})} k_r \cdot \mathbf{v}_r \text{ for some } k_r \in \mathbb{Q}^+. \tag{1}$$

Proof: Let P be a probability measure that satisfies all CI statements in $\mathcal{C}$ and let ΔM_P be the Möbius inversion of the multi-information function M_P induced by P. Since M_P is a supermodular function [**46**], we have $\sum_{r \in \mathcal{R}(\mathcal{C})} (k_r \cdot \sum_{U \in \mathcal{L}(r)} \Delta M_P(U)) \geq 0$, and also $\sum_{U \in \mathcal{L}(c)} \Delta M_P(U) \geq 0$. Now, since P satisfies all CI statements in $\mathcal{C}$ we have that $\sum_{c' \in \mathcal{C}} \sum_{U \in \mathcal{L}(c')} \Delta M_P(U) = 0$ by Lemma 5.20. Since equality (1) holds by assumption, we have that

$$\sum_{c' \in \mathcal{C}} \sum_{U \in \mathcal{L}(c')} \Delta M_P(U) = \sum_{U \in \mathcal{L}(c)} \Delta M_P(U) + \sum_{r \in \mathcal{R}(\mathcal{C})} (k_r \cdot \sum_{U \in \mathcal{L}(r)} \Delta M_P(U)) = 0.$$

Hence, $\sum_{U \in \mathcal{L}(c)} \Delta M_P(U) = 0$ and, by Lemma 5.20, it follows that P satisfies c.

In light of these results, we can now rewrite equation (1) in the previous proposition as a linear program (Schrijver [**42**]). A linear program has the form

$$\text{minimize } \mathbf{c}^T\mathbf{x} \tag{2}$$

$$\text{subject to } \mathbf{A}\mathbf{x} \text{ eq } \mathbf{b}, \mathbf{x} \geq 0 \tag{3}$$

where eq is one of $\{\leq, \geq, =\}$. For our purposes, eq is the equality sign, the columns of matrix $\mathbf{A}$ are the vectors $\mathbf{v}_r$ for each of the relevant elementary CI statements, that is, for the CI statements in $\mathcal{R}(\mathcal{C})$, and $\mathbf{b} = \mathbf{v}_{\mathcal{C}} - \mathbf{v}_c$. Clearly, our objective function is the zero-function $\mathbf{0}^T$ because we are only interested in the existence of a solution for the equality constraints. This is often referred to as the *feasibility problem* of finding a solution for the system of linear constraints.

Example 5.23. Let $S = \{a, b, c, d\}$ and let $\mathcal{C} = \{I(a, b|\emptyset), I(c, d|a), I(c, d|b), I(a, b|cd)\}$. Then, $\mathcal{R}(\mathcal{C}) = \{I(a, b|\emptyset), I(a, b|c), I(a, b|d), I(a, b|cd), I(c, d|\emptyset),\ I(c, d|a), I(c, d|b), I(c, d|ab)\}$ and $\mathcal{L}(\mathcal{C}) = \{\emptyset, a, b, c, d, ab, cd\}$. The columns $\mathbf{e}_1, ..., \mathbf{e}_8$ of the *minimal* 0-1 matrix $\mathbf{A}$ below correspond to the eight *relevant* elementary CI statements and the number of rows is determined by $\mathcal{L}(\mathcal{C})$.

$$\mathbf{A} = \begin{array}{c|cccccccc|} & \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 & \mathbf{e}_4 & \mathbf{e}_5 & \mathbf{e}_6 & \mathbf{e}_7 & \mathbf{e}_8 \\ \hline \text{cd} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ \text{ab} & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \text{a} & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ \text{b} & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ \text{c} & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \text{d} & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \emptyset & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}$$

We have that $\mathbf{v}_{\mathcal{C}}^T = (2, 2, 1, 1, 1, 1, 1)$. Now, let $I(c, d|\emptyset)$ be a CI statement. Then we have that $\mathbf{b}^T = \mathbf{v}_{\mathcal{C}}^T - \mathbf{v}_{I(c,d|\emptyset)}^T = (2, 2, 1, 1, 1, 1, 1) - (0, 1, 1, 1, 0, 0, 1) = (2, 1, 0, 0, 1, 1, 0)$. Finally, it follows that $\mathcal{C} \models I(c, d|\emptyset)$, since $\mathbf{b} = \mathbf{e}_2 + \mathbf{e}_3 + \mathbf{e}_8$.

It is well-known that linear programs (LPs) are solvable in polynomial time in the number of variables. However, in the worst case the reduction leads to an LP with an exponential number of variables ($\binom{|S|}{2}2^{|S|-2}$; the maximum number of elementary CI statements over S) and constraints ($2^{|S|} - |S| - 1$; the maximum cardinality of the set $\mathcal{L}(\mathcal{C})$). As a rule of thumb, the more columns matrix $\mathbf{A}$ has the more difficult is the corresponding LP problem. An advantage of our method over a naïve approach is that $\mathbf{A}$ only consists of the vectors representing the *relevant* elementary CI statements $\mathcal{R}(\mathcal{C})$. This means that the number of columns (that is, the number of variables of the LP) can be very small compared to the worst case. In rare cases, the solutions to the LPs might be inaccurate due to round-off and truncation errors. Therefore, when we obtain a solution, we expand the elements of the solution vector into fractions of integers, which results in a vector $\mathbf{x}_f$, and only accept if $\mathbf{A}\mathbf{x}_f = \mathbf{b}$. We also would like to underscore that matrix $\mathbf{A}$ is always a 0-1 matrix, leading to better numerical stability and the possibility to take advantage of existing sparse matrix data structures. We will come back to algorithmic issues when we discuss the results of our experiments.

6.1. Combinatorial and Structural Imsets. There is a close link to Studený's theory of imsets [**46**], on which we will briefly elaborate in this section. Let $\mathcal{C}$ be a set of CI statements and let c be a CI statement over a set S. Then, under the assumption that we can ignore numerical inaccuracies, one can test whether imset $u_{\mathcal{C}} - u_c$ is structural using the previously introduced reduction to a linear program. Furthermore, one can test whether the imset is combinatorial by reducing it to the identical integer program. Again, we want to stress that numerical rounding and truncation errors might lead to inaccurate results, and, therefore, the method should be used with caution when mathematical properties about combinatorial and structural imsets are to be proved. However, one of the results of our experiments is that the solver of the LP instances delivered integer and small rational solutions in all but some cases which allowed us to verify their correctness. We refer the reader to Hemmecke et al. [**18**] who used, among other tools, integer programming to find a structural imset which is not combinatorial.

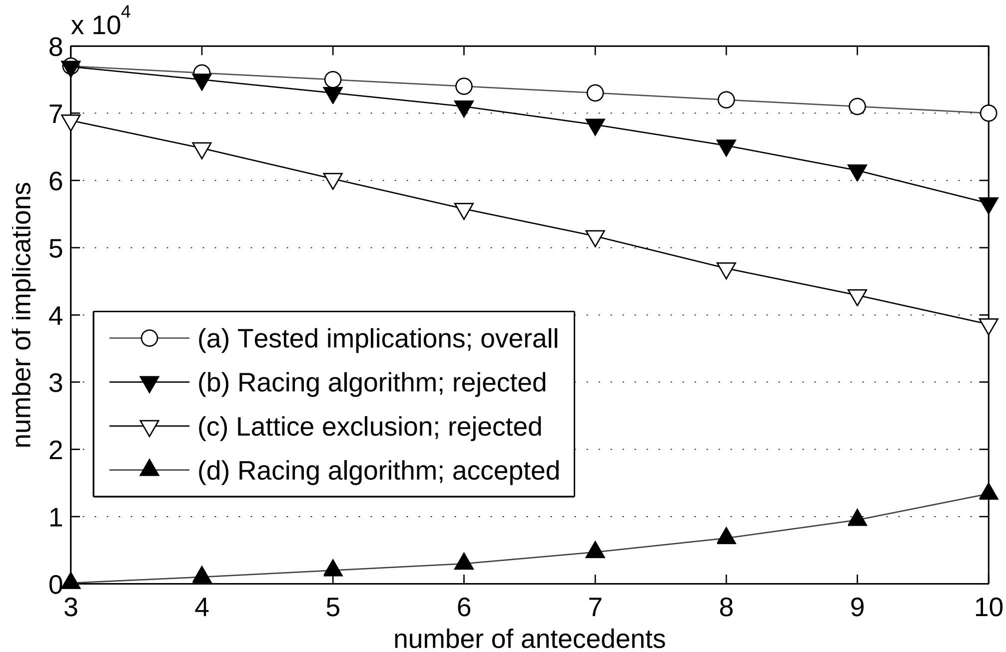

FIGURE 5.4. Rejection and acceptance curves of the racing and falsification algorithms, respectively, for five attributes.

7. Experiments

7.1. Falsification Algorithm and Heuristics. With our experiments we want to show that (1) the lattice-exclusion criterion can falsify a large fraction of all falsifiable implications, and (2) that the two provided heuristics are good approximations of the full-blown lattice-exclusion criterion. To make our outcomes comparable to existing results, we adopted the experimental setup for the *racing algorithm* from Bouckaert and Studený [**3**] (also using 5 attributes). We first randomly selected 1,000 times 3 up to 10 elementary CI statements, resulting in a total of 8,000 sets of antecedents.[2] The *falsification algorithm* and the heuristics were run on these sets with each of the remaining elementary CI statements as a consequence, one at a time. Since there are 80 elementary CI statements for 5 attributes, this resulted in 77,000 implication problems for sets with 3 antecedents, 76,000 for sets with 4 antecedents, down to 70,000 for sets with 10 antecedents.

[2]An elementary CI statement is of the form $I(a, b|C)$, where $a, b \in S$ and $C \subseteq S - \{a, b\}$.

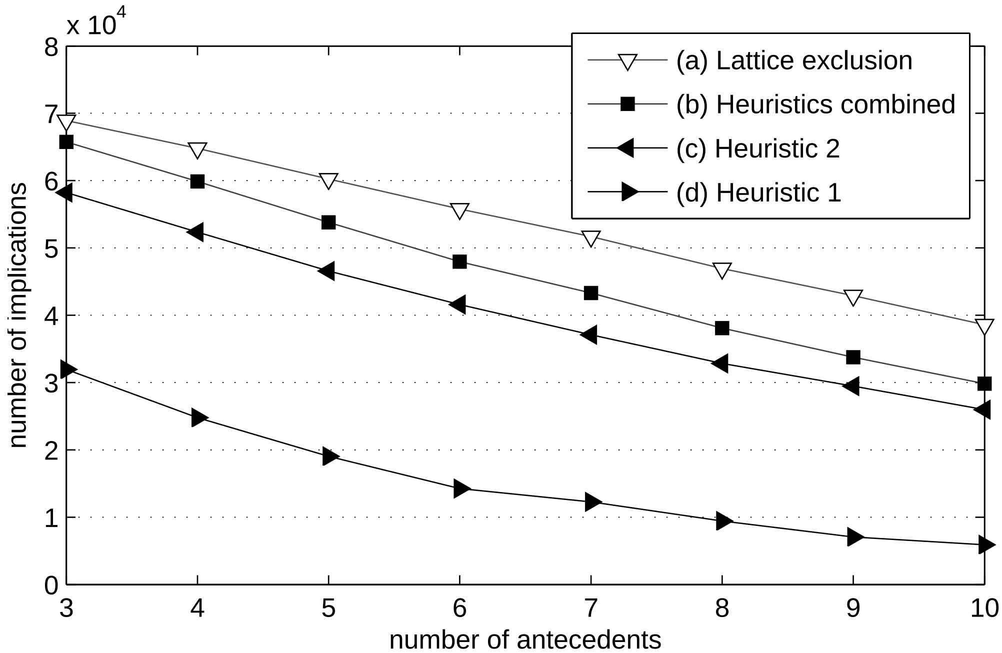

FIGURE 5.5. Falsifications based on the lattice-exclusion criterion and the heuristics, for five attributes. The combination of the heuristics yields 95% of the falsifications of the full-blown lattice exclusion criterion for 3 antecedents down to 77% for 10 antecedents.

The rejection procedure of the *racing algorithm* is rooted in the theory of imsets: an instance is rejected if one of the supermodular functions constructed by the algorithm is a counter-model for this instance. It has exponential running time and might reject implications that actually *do* hold (false negatives). This is a consequence of the fact that $\mathcal{M}$ is a *strict* subset of the class of all supermodular functions. (See Examples 4.1 and 6.2 in Studený's monograph [**46**].) The *falsification algorithm* based on Corollary 5.16, on the other hand, ensures that if an instance of the implication problem is rejected, then it is guaranteed not to be valid.

Figure 5.4 shows the rejection curves of the *racing algorithm* (b) and the *falsification algorithm* (c), respectively, and the acceptance curve of the racing algorithm (d). The area between the two rejection curves can be interpreted as the "decision gap", i.e., the amount of instances of the implication problem for which the validity is unknown. The curve marked

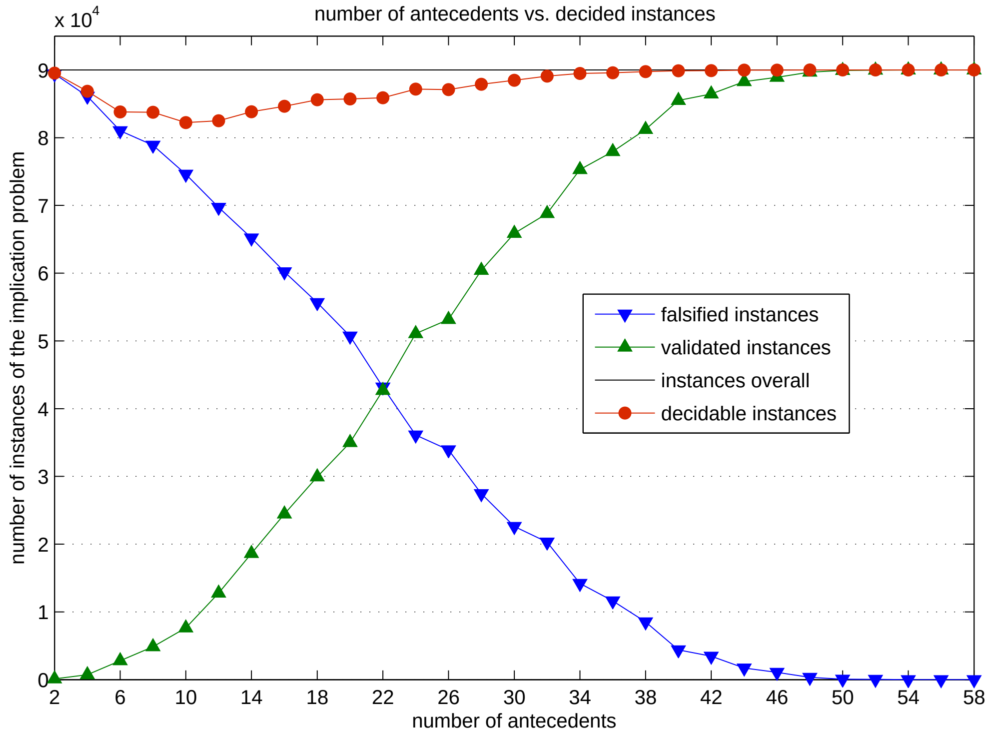

FIGURE 5.6. Falsification and validation curves of the approximate decision algorithm for five variables. The curve marked with circles depicts the number of instances which we could either falsify or validate.

with circles (a) depicts the total number of tested instances. Figure 5.5 depicts the rejection curves for the *falsification algorithm* (a), for the combination of Heuristic 1 and Heuristic 2 (b), and for Heuristic 2 (c) and Heuristic 1 (d) run separately. The combination of the heuristics compares favorably to the full-blown *falsification criterion*. The experiments also show that Heuristic 2 is more effective than Heuristic 1.

7.2. Validation Algorithm. We will mainly address the following empirical questions with our experiments:

(1) **Effectiveness**: What fraction of the instances of the implication problem can we either falsify or validate?

(2) **Efficiency**: How fast does the algorithm run and to how many variables does it scale? How much more efficient is the algorithm compared to the naïve approach both in terms of time and space complexity?

(3) **Structural and numerical properties**: How large is the constraint matrix $\mathbf{A}$ for different instances? What are the numerical properties of the solutions?

To judge the effectiveness and efficiency of the algorithm, we must apply it to instances of the implication problem over different number of variables. Since the distribution of implication problems in real-world applications is unknown, our experiments need to be run on randomly generated instances. Using the method of randomly generating test instances from (Bouckaert and Studený [**3**]) allows us to compare the experimental outcomes with existing results. Hence, for each experiment, we first generated instances of the implication problem $(S, \mathcal{C}, c)$ by randomly selecting n different sets of elementary CI statements over S as antecedents $\mathcal{C}$, and for each of these, k different elementary CI statements c over S as consequence, one at a time. We first applied the falsification algorithm to these instances and used the remaining CI statements to create the constraint matrix $\mathbf{A}$ and vector $\mathbf{b}$ from $\mathcal{C}$ and c as described in the previous section. For the resulting linear programs we used lp_solve[3] an open-source linear programming system that can solve both linear and integer programs. It is based on the revised simplex method and the branch-and-bound method for integer programs. We did not change the standard optimization settings of the solver. Furthermore, we only accepted a solution if its rational expansion solved the respective constraints. For our purposes this is unproblematic because the objective is to validate as many instances of the implication problem as possible while entirely ruling out false positives. All experiments were run on a dual-core 3.2GHz Linux PC with 2GB RAM.

Figure 5.6 shows the number of instances that could either be validated or falsified by the algorithms for five variables. For each $\ell = 2, \ldots, 58$ (the number of antecedents) we randomly created 4,500 different sets of ℓ elementary CI statements, and for each of these

[3]Michel Berkelaar, Kjell Eikland, and Peter Notebaert. lp_solve, an open source (Mixed-Integer) Linear Programming system originally developed at Eindhoven University of Technology; http://lpsolve.sourceforge.net/

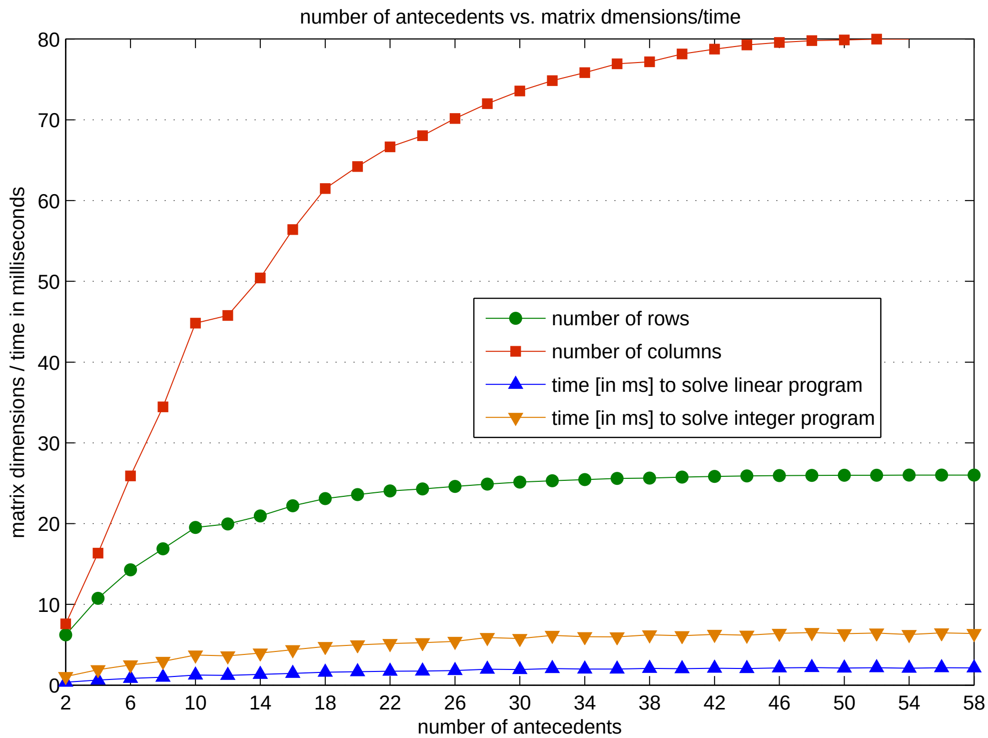

FIGURE 5.7. Dimensions of matrix $\mathbf{A}$ that encodes a set of x antecedents over five variables; and time in milliseconds needed to solve the corresponding linear and integer programs.

randomly selected 20 different elementary CI statements as consequences, one at a time, resulting in 90,000 instances of the implication problems for each ℓ. The results show that only a small fraction of the instances could not be decided and that for larger values of ℓ (for five variables: $\ell > 40$) all of the instances could either be falsified or validated. This behavior of the algorithm was consistent over all tested number of variables $(4, \ldots, 15)$.

Figure 5.7 depicts the average size (that is, the number of columns and rows) of the constraint matrix $\mathbf{A}$, and the average time in milliseconds to solve one linear program and the corresponding integer program, respectively. After solving the linear programs, that is, determining whether or not there exists a solution, we also solved the equivalent integer programs. Interestingly, for each and every of the 2,700,000 instances for five variables, if there existed a solution to the linear program, then there also existed one for the corresponding integer program.

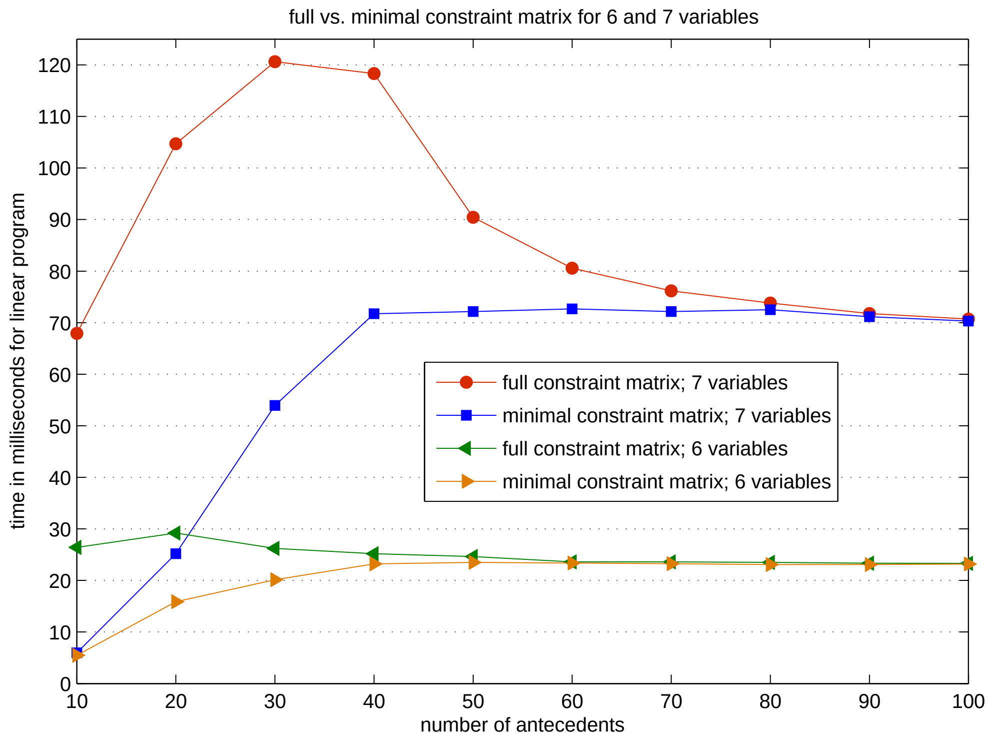

FIGURE 5.8. Average time needed (in ms; 30,000 trials) to solve a linear program with and without optimizing the constraint matrix **A**; average of 30,000 trails for six and seven variables, respectively.

Figure 5.8 illustrates the computational advantage one gains when using the *minimal* constraint matrix **A** consisting only of the *relevant* elementary CI statements, over using the matrix consisting of all elementary CI statements. The times in milliseconds provided are for 6 and 7 variables, averaged over 30,000 trials, for 1000 sets of $\ell = 10, 20, \ldots, 100$ antecedents, and 30 different consequences, one at a time. Figure 5.9 depicts the average time in seconds to solve instances of the linear programs and the average dimensions of constraint matrix **A** for different number of variables, averaged over 1000 trails.

Finally, we want to compare our algorithm to the *racing algorithm* introduced in (Bouckaert and Studený [**3**]). The falsification procedure of the racing algorithm is rooted in the theory of imsets: an instance of the implication problem is falsified if one of the supermodular functions constructed by the algorithm is a counter-model for the instance of the implication problem [**3**]. It is heavily randomized, has super-exponential running time, and

variables	time [s]	rows(**A**)	columns(**A**)
6	0.024	57 (57)	239 (240)
7	0.073	117 (120)	592 (672)
8	0.642	230 (247)	1193 (1792)
9	1.580	423 (502)	1852 (4608)
10	2.647	687 (1013)	2422 (11520)
11	7.316	1221 (2036)	3699 (28160)
12	9.038	2039 (4083)	4786 (67582)
13	20.267	3331 (8178)	6863 (159744)
14	35.969	4986 (16369)	8298 (372736)
15	91.237	6713 (32752)	11024 (860160)

FIGURE 5.9. The values are the average time (in seconds) needed to solve the linear program, and the average number of rows and columns of the constraint matrix **A**; out of 1000 trials with 50 antecedents. The values in parentheses are the maximal possible values.

could therefore only be tested for up to 6 variables. Furthermore, the racing algorithm might falsify implications that actually do hold. This is a consequence of the fact that the class of multi-information functions induced by discrete probability measures is a strict subset of the class of all supermodular functions. (See Examples 4.1 and 6.2 in Studený's monograph [**46**].) The *falsification algorithm* based on Proposition 5.16 ensures that falsified instance of the implication problem are guaranteed not to be valid. The validation procedure of the racing algorithm tests whether an imset that *encodes* an instance of the implication problem is combinatorial. It makes use of some ad-hoc heuristics to speed-up the computations. The validation algorithm presented here introduces two novel ideas: (1) the representation of instances of the implication problem as linear programs; and (2) the notion and construction of *minimal* constraint matrices that increase the efficiency of the algorithm.

CHAPTER 6

Logical and Algorithmic Properties of Stable Conditional Independence

We have seen that conditional independence plays a fundamental role for learning and reasoning in intelligent systems. A conditional independence (CI) statement speaks to the independence of two sets of random variables relative to a third: given three mutually disjoint sets A, B, and C of random variables, A and B are conditionally independent relative to C if any instantiation of the variables in C renders the variables in A and B independent. In other words, if we have knowledge about the state of C, then knowledge about the state of A does not provide additional evidence for the state of B and vice versa. We use the notation $I(A, B|C)$ to specify this independence condition.

When novel information becomes available in a probabilistic system, the set of associated, relevant CI statements changes dynamically. However, some of the CI statements will continue to hold, i.e., they remain *stable* under change in the system. Technically, the notion of stability of a CI statement $I(A, B|C)$, in the context of a set of random variables S and a set of CI statements $\mathcal{C}$, is defined by requiring that, for every superset $C' \supseteq C$ which is disjoint from A and B, the CI statement $I(A, B|C')$ also holds. In other words, the independence of A and B relative to C is unaffected by adding random variables to C. Clearly, this property does not hold in general. Adding variables to the set C may affect the (in-)dependence of A and B. A special case for which the stability of the CI statement $I(A, B|C)$ is guaranteed is the situation where $A \cup B \cup C = S$. (Recall that, when $A \cup B \cup C = S$, the CI statement $I(A, B|C)$ is said to be *saturated*.)

Among the most frequently used models for representing conditional independence information are *graphs*, wherein the nodes correspond to random variables and the edges encode the (in-)dependence information among the variables. There are three main types

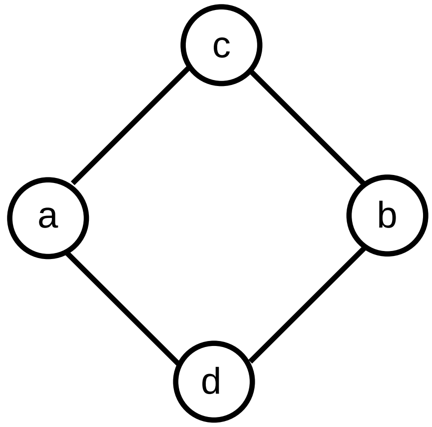

FIGURE 6.1. An undirected graphical model over 4 variables representing the stable CI structure $\{I(a,b|cd), I(c,d|ab)\}$. Please note that we always omit symmetric and trivial CI statements.

of graphical models: undirected graphs, directed graphs, and chain graphs. We focus specifically on undirected graphical models (also called Markov networks), since we will show that the class of stable CI structures is a strict generalization of the class of CI structures represented by Markov networks. Let $S = \{a, b, c, d\}$ be a set of random variables, and let G be the Markov network shown in Figure 6.1. Then, G represents the CI statements $I(a, b|cd)$ and $I(c, d|ab)$. (We will write $a_1 \cdots a_n$ to denote the set $\{a_1, \ldots, a_n\}$.)

One of the useful properties of the existence of a stable CI statement $I(A, B|C)$ in a set of CI statements $\mathcal{C}$ is that, in a representation of $\mathcal{C}$, it is not necessary to further represent CI statements of the form $I(A, B|C')$, where C' is a strict superset of C. This can lead to a substantial decrease in the number of CI statements that need to be maintained in the system. The importance of stable conditional independence for reducing the complexity of representation of conditional independence structures has recently been established [**9**].

We approach the paradigm of stable CI as a strict generalization of Markov networks to represent and reason about conditional independence. We believe that a good understanding of its logical and algorithmic properties will lead to new theoretical insights and applications in the field of uncertain reasoning. While several results regarding these properties exist [**9, 27, 10**], no study has investigated these as comprehensively as it was done for unrestricted CI and graphical models relative to the class of discrete probability measures [**15**]. We extend this study to stable conditional independence by utilizing recent results concerning a finite sound and complete axiomatization of the implication problem for stable CI statements, relative to discrete probability measures [**30**]. In particular, we show that (1) every stable CI structure has a perfect model, i.e., a discrete probability measure that satisfies precisely the CI statements in $\mathcal{C}$, (2) the number of distinct stable CI structures grows at least double-exponentially with the number of random variables, and (3) every set of CI statements represented by a Markov network is a set of stable CI statements. We establish a direct connection between sets of stable CI statements and propositional formulas in conjunctive normal form and use this connection to show that the conditional independence implication problem for stable conditional independence is coNP-complete. In light of these results, we present experimental results that show how existing SAT solvers can be used to (1) decide instances of the stable CI implication problem and (2) compute concise, non-redundant representations of stable CI structures, even for instances involving hundreds of random variables.

1. Conditional Independence and Markov Models

Throughout this chapter, S will be a non-empty finite set of random variables.

Definition 6.1. A Markov network over a finite set S is an undirected graph G with nodes corresponding to random variables in S. The conditional independence statement $I(A, B|C)$ is represented by G if every path in G between a node in A and a node in B contains a node in C, or, equivalently, if C separates A and B.

Each Markov network G over S represents a set of conditional independence statements through this separation criterion. The set of CI statements *represented* by G will be denoted by $\mathcal{C}(G)$. Every set of CI statements $\mathcal{C}(G)$ *represented* by a Markov network G will be called a *Markov model*.

Markov models can be completely axiomatized using the inference system in Figure 6.3 [**32**]. The Markov network in Example 6.1 represents the set of CI statements

$$\{I(a, b|cd), I(c, d|ab)\}.$$

Please note that we always omit symmetric and trivial CI statements.

In this section, we recall the notion of a probability measure satisfying a CI statement and apply it in the context of Markov networks.

Definition 6.2. Let S be a finite set of random variables. A probability measure P is Markovian with respect to a Markov network G over S, if $I(A, B|C)$ is represented by G implies that P satisfies $I(A, B|C)$. A probability measure P is perfectly Markovian with respect to G if the converse implication holds as well.

We can now define the notion of perfect models for sets of CI statements and the notion of a CI structure.

Definition 6.3. Let $\mathcal{C}$ be a set of CI statements over S. $\mathcal{C}$ is a *CI structure* if and only if $\mathcal{C} = \mathcal{C}^*$. Furthermore, we say that a probability measure P is a perfect model for $\mathcal{C}$ if P satisfies precisely the CI statements in $\mathcal{C}^*$.

2. Inference Systems for the Probabilistic Conditional Independence Implication Problem

Given the notion of a CI implication problem, it is common place to consider inference rules and inference systems that are *sound* for these problems. An inference rule (an inference system) is sound relative to the class of discrete probability measures if it infers, given a set of CI statements $\mathcal{C}$, only CI statements in $\mathcal{C}^*$. If an inference system can infer all CI statements in $\mathcal{C}^*$, it is said to be complete.

The best know sound inference system for the CI implication problem relative to the class of discrete probability measures is the *semi-graphoid axiom system* [**8, 31**]. We denote it by $\mathcal{G}$ and its inference rules are depicted in Figure 5.1. Note, however, that system $\mathcal{G}$ is not complete. In fact, it is known that there does not exist a finite set of sound inference rules

$I(A, \emptyset \mid C)$	**Triviality**
$I(A, B \mid C) \rightarrow I(B, A \mid C)$	**Symmetry**
$I(A \cup D, B \mid C)$ & $I(A, D \mid B \cup C) \rightarrow I(A, B \cup D \mid C)$	**Contraction**
$I(A, B \cup D \mid C) \rightarrow I(A, B \mid C \cup D)$	**Weak union**

FIGURE 6.2. System $\mathcal{S}$ is sound and complete for the CI implication problem for saturated statements. Note that the inference rule **contraction** has a slightly different form to accommodate saturated CI statements [**15**].

that is sound and complete for the implication problem on unrestricted CI statements [**46**]. It is also unknown whether this implication problem is decidable.

For the implication problem for saturated CI statements, the situation is different. In Figure 6.2, system $\mathcal{S}$ is shown, which is a finite set of inference rules that is sound and complete for this implication problem relative to the class of discrete probability measures [**32**].

For sets of CI statements represented by Markov networks, the situation is yet different. Figure 6.3 depicts system $\mathcal{M}$, which is a finite set of inference rules that is sound and complete for the implication problem for sets of CI statements represented by Markov networks, relative to the class of discrete probability measures [**32**].

Let $\mathcal{I}$ be an inference system for CI statements. As before, the derivability of a conditional independence statement c from a set of conditional independence statements $\mathcal{C}$ under the inference rules of system $\mathcal{I}$ is denoted by $\mathcal{C} \vdash_{\mathcal{I}} c$. The *closure* of $\mathcal{C}$ under $\mathcal{I}$, denoted $\mathcal{C}^{\mathcal{I}+}$, is the set $\{c \mid \mathcal{C} \vdash_{\mathcal{I}} c\}$.

3. Stable Conditional Independence

When novel information becomes available to a probabilistic system, the set of associated, relevant CI statements changes dynamically. However, some of these CI statements will continue to be satisfied, i.e., they remain *stable*. The paradigm of stable conditional independence, and some of its the properties, were first investigated by Matúš [**27**], who

$I(A, \emptyset \mid C)$	**Triviality**
$I(A, B \mid C) \rightarrow I(B, A \mid C)$	**Symmetry**
$I(A, B \cup D \mid C) \rightarrow I(A, D \mid C)$	**Decomposition**
$I(A, B \mid C) \rightarrow I(A, B \mid C \cup D)$	**Strong union**
$I(A, B \mid C \cup D)$ & $I(A, D \mid B \cup C) \rightarrow I(A, B \cup D \mid C)$	**Intersection**
$I(A, B \mid C) \rightarrow I(A, \{d\} \mid C) \vee I(\{d\}, B \mid C)$	**Transitivity**

FIGURE 6.3. System $\mathcal{M}$ is sound and complete for the CI implication problem for CI statements represented by Markov networks.

named it *ascending* conditional independence, and later by de Waal and van der Gaag [**9**], who coined the term *stable* conditional independence. Every set of CI statements can be partitioned into its *stable* and *unstable* part. In this section, we will recall two different characterizations of stable CI structures, one using a finite set of inference rules, and the other using the *lattice-inclusion* property of CI statements [**30**]. We will harness these results to prove several important properties about stable CI. The set of inference rules in Figure 5.2 will be denoted by $\mathcal{A}$. The *symmetry*, *decomposition*, and *contraction* rules are part of the semi-graphoid axioms [**8, 31**] (see Figure 5.1). *Strong union* and *strong contraction* are additional inference rules.

Stable independence can be defined *relative* to a set of CI statements [**9, 10**]. However, we approach the paradigm of stable CI as a mechanism for the succinct representation of conditional independence information, much like graphical models are used for this purpose. Instead of assuming that every CI statement is satisfied by a probability measure which is perfectly Markovian with respect to a graphical model, we assume that every CI statement is satisfied by a probability measure which is perfectly Markovian with respect to a set of stable CI statements. Therefore, a set of stable conditional independence statements will be *any* set of CI statements that are *implicitly known* (i.e., assumed) to be stable. Whenever we say that a set of CI statements is stable, we implicitly assume that $\mathcal{C}^*$ satisfies the required condition. Hence, in general, a set of stable CI statements $\mathcal{C}$ can be different from the set $\mathcal{C}^*$. The motivation for this approach is to achieve a structural representation of conditional independence information which is broader than Markov networks but still allows for efficient implication testing and storage. The next definition formalizes this approach.

Definition 6.4. Let $\mathcal{C}$ be a set of CI statements. We say that $\mathcal{C}$ is a set of stable CI statements if, for all $I(A, B|C) \in \mathcal{C}$ and for all $C' \supseteq C$, we have that $I(A, B|C') \in \mathcal{C}^*$. Equivalently, a set of stable CI statements is a set of CI statements for which the inference rule strong union (see Figure 6.3) is sound. A *stable CI structure* is a set of stable CI statements $\mathcal{C}$ such that $\mathcal{C} = \mathcal{C}^*$.

The following result follows immediately from this definition.

proposition 6.5. *Let $\mathcal{C}$ be a set of saturated CI statements over S. Then $\mathcal{C}$ is a set of stable CI statements over S.*

In analogy to the definition of a (perfectly) Markovian probability measure with respect to graphical models (Definition 6.2), we can define the concept of a (perfectly) Markovian probability measure with respect to stable CI structures.

Definition 6.6. Let $\mathcal{C}$ be a stable CI structure. A probability measure P is Markovian with respect to $\mathcal{C}$ if $I(A, B|C) \in \mathcal{C}$ implies that P satisfies $I(A, B|C)$. A probability measure P is perfectly Markovian with respect to $\mathcal{C}$ if the converse implications holds as well. We say that a probability measure is a perfect model for $\mathcal{C}$ if and only if it is perfectly Markovian with respect to $\mathcal{C}$.

3.1. The Implication Problem for Stable Conditional Independence. Here, we recall two characterizations of the implication problem for stable CI statements (the stable CI implication problem), one in terms of a finite system of inference rules, and another using the *lattice-inclusion property* [**30**]. We will use these results to show that each stable CI structure has a perfect model with respect to discrete probability measures, but not with respect to binary discrete probability measures.

A powerful tool in deriving results about the CI implication problem is the association of semi-lattices with CI statements [**30**]. Given subsets A and B of S, we write $[A, B]$ for the lattice $\{U \mid A \subseteq U \subseteq B\}$.

We can now state the two characterizations for the conditional independence implication problem for stable CI statements relative to the class of discrete probability measures (see also [**30**]).

THEOREM 6.7. *Let $\mathcal{C}$ be a set of stable CI statements over S and let c be a CI statement over S. Then, the following statements are equivalent:*

(a) $\mathcal{C} \models c$;

(b) $\mathcal{C} \vdash_{\mathcal{A}} c$; *and*

(c) $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(c)$.

Proof: We know that $\mathcal{M}$ has the zero-density property with respect to stable CI statements since **strong union** is a sound inference rule for the stable CI implication problem, by the definition of stable conditional independence, and **decomposition** is a sound inference rule for unrestricted CI statements (Geiger and Pearl [**15**]). Hence, by Theorem 3.13, $\mathcal{A}$ is a sound inference system for the stable CI implication problem, and we have shown that (b) implies (a). Since $\mathcal{M}$ has the Kronecker property on $\mathcal{S}^2(S)$, we know that inference system $\mathcal{A}$ has to be complete for the stable CI implication problem, and it follows that (a) implies (b). Finally, we have that (b) if and only if (c) by Theorem 3.10.

EXAMPLE 6.8. Let $S = \{a, b, d, e\}$, let $\mathcal{C} = \{I(a, b|\emptyset), I(d, e|a), I(d, e|b)\}$ be a set of stable CI statements, and let $c = I(d, e|\emptyset)$. We know, by *strong contraction*, that $\mathcal{C} \vdash_{\mathcal{A}} c$ and, therefore, $\mathcal{C} \models c$ by Theorem 6.7. Now, $\mathcal{L}(\mathcal{C}) = \{\emptyset, d, e, de\} \cup \{a, ab\} \cup \{b, ab\} = \{\emptyset, a, b, d, e, ab, de\} \supseteq \{\emptyset, a, b, ab\} = \mathcal{L}(c)$.

One might expect that, based on the definition of stable CI, it would be sufficient to have the inference rule *strong union* in system $\mathcal{A}$ but not *strong contraction*. However, as the following example demonstrates, system $\mathcal{A}$ without *strong contraction* is *not* complete for the stable CI implication problem.

EXAMPLE 6.9. Let $S = \{a, b, c, d\}$ and consider the set of stable CI statements $\mathcal{C} = \{I(a, b|\emptyset), I(a, b|c), I(a, b|d), I(a, b|cd), I(c, d|a), I(c, d|b), I(c, d|ab)\}$. We know that $I(c, d|\emptyset)$ is implied by $\mathcal{C}$ [**43**]. However, one can verify that $I(c, d|\emptyset)$ cannot be derived from $\mathcal{C}$ under $\mathcal{A}$ without the rule *strong contraction*.

The next result follows from the existence of discrete perfect models with respect to CI statements [**15**], a result which was later strengthened by [**34**].

PROPOSITION 6.10. *For every stable CI structure $\mathcal{C}$, there exists a discrete probability measure P such that P is a perfect model for $\mathcal{C}$.*

However, the previous result does not hold for the class of binary discrete probability measures.

PROPOSITION 6.11. *There exists a stable CI structure for which no binary discrete probability measure is a perfect model.*

Proof: Let $S = \{a, b, c\}$ and let $\mathcal{C} = \{I(a, b|\emptyset), I(a, b|c)\}$. We have that $\mathcal{L}(\emptyset, \{\{a\}, \{b\}\}) = \{\emptyset, c\}$ and $\mathcal{L}(c, \{\{a\}, \{b\}\}) = \{c\}$. Now, since $\mathcal{L}(\emptyset, \{\{a\}, \{c\}\}) = \{\emptyset, b\}$ and $\mathcal{L}(\emptyset, \{\{b\}, \{c\}\}) = \{\emptyset, a\}$, we have, by Theorem 6.7 (c), that neither $I(a, c|\emptyset)$ nor $I(b, c|\emptyset)$ are implied by $\mathcal{C}$. Hence, $\mathcal{C}$ is a stable CI structure. However, we know that every binary probability measure that satisfies the CI statements in $\mathcal{C}$ also satisfies either $I(a, c|\emptyset)$ or $I(b, c|\emptyset)$ [**15**]. Thus, no binary probability measure is a perfect model for $\mathcal{C}$.

The combination of these result shows that the paradigm of stable CI has the same perfect model properties as unrestricted CI.

3.2. Markov Networks and Stable Conditional Independence. The primary goal of this section is to relate stable conditional independence to Markov networks. In particular, we will show that every set of CI statements represented by a Markov network is a stable CI structure. Consequently, Markov networks are a specialization of the more general notion of stable conditional independence.

THEOREM 6.12. *Let G be a Markov network over S. Then, the set of CI statements represented by G, that is, $\mathcal{C}(G)$, is a stable CI structure.*

Proof: It is well-known that *strong union* is a sound inference rule for separation in undirected graphs [**31**] (see Figure 6.3). In addition, it can be verified that the inference rule *strong contraction* is sound for separation in undirected graph. Thus, inference system $\mathcal{A}$ is sound for separation in Markov networks and the statement of the theorem follows.

COROLLARY 6.13. *For every Markov network G there exists a stable CI structure $\mathcal{C}$, and every discrete probability measure that is (perfectly) Markovian w.r.t. G satisfies the elements in $\mathcal{C}$ (and none other).*

Theorem 6.12 implies that the notion of stable conditional independence is a generalization of Markov networks. In what follows, we will investigate how much broader this notion is compared to such networks. First, we show that there exists a stable CI structure that cannot be represented by a Markov network.

PROPOSITION 6.14. *There exists a stable CI structure $\mathcal{C}$ over a set S such that for each Markov network G over S, $\mathcal{C} \neq \mathcal{C}(G)$.*

Proof: Let $S = \{a, b, c, d\}$ and let $\mathcal{C} = \{I(a, b|cd), I(a, d|bc)\}$ be a set of stable CI statements. We have that $\mathcal{L}(cd, \{\{a\}, \{b\}\}) = \{cd\}$ and $\mathcal{L}(bc, \{\})\{a\}, \{d\} = \{bc\}$. Hence, by Theorem 6.7 (c), no other CI statement is implied by $\mathcal{C}$. Thus, $\mathcal{C}$ is a stable CI structure. However, every Markov network that represents these two CI statements also represents the CI statement $I(a, bd|c)$ by the inference rule *intersection* (see Figure 6.3) which is sound for separation in undirected graphs [**31**]. Thus, the class of all CI structures represented by the class of Markov networks is a strict subclass of the class of stable CI structures.

Figure 6.4 depicts some relationships between different structural representations of conditional independence information. Please note that each saturated CI structure is trivially a stable CI structure.

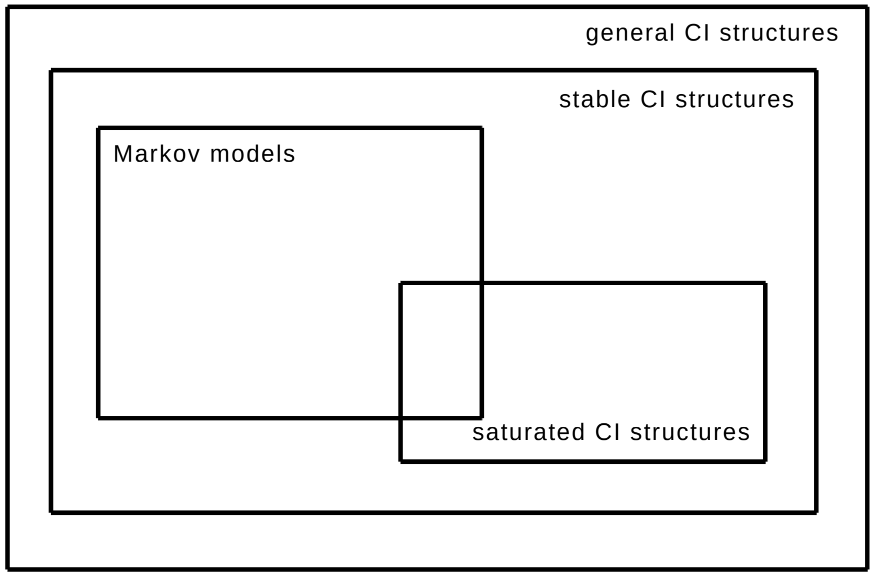

FIGURE 6.4. Inclusion relationships between different representations of conditional independence. Every Markov model is a stable CI structure (Theorem 6.12). Every saturated CI structure is trivially a stable CI structure.

3.3. Some Combinatorics about Stable Conditional Independence.

In this section, we will show, given a set of random variables S, that the number of distinct stable conditional independence structures grows at least double-exponentially. This shows analytically that stable conditional independence can represent a much broader class of CI structures compared to undirected models, since there can only be $2^{(|S|(|S|-1))/2}$ different undirected graphical models over a set of random variables S.

LEMMA 6.15. *Let S be a finite set of discrete random variables. Then, the number of distinct stable CI structures d_S over S is at least*

$$d_S \geq \sum_{i=0}^{|S|-2} (2^{\binom{|S|}{2}\binom{|S|-2}{i}} - 1).$$

Proof: Let S be a finite set, let $V \subseteq S$ with $|V| = |S|-2$, and let $U \subseteq V$. For every lattice $[U, V]$, there exists a stable CI structure $\mathcal{C}$ (for instance, $\mathcal{C} = \{I(u, v|U') \mid U' \supseteq U, \{u, v\} =$

$S - V\})$ such that $\mathcal{L}(\mathcal{C}) = [U, V]$. Consider the set $D_i^S = \{[U, V] \mid |V| = |S| - 2, |U| = |S| - 2 - i, U \subseteq V \subseteq S\}$. There are $\binom{|S|}{2}$ different subsets of S of size $|S| - 2$. Each of these subsets V has $\binom{|S|-2}{|S|-2-i} = \binom{|S|-2}{i}$ different subsets of size $|S| - 2 - i$. Hence, we have that $|D_i^S| = \binom{|S|}{2}\binom{|S|-2}{i}$. Now, for every $i = 0...(|S| - 2)$, each non-empty subset of the set D_i^S corresponds to a set of stable CI statements whose union of semi-lattices is distinct from the union of semi-lattices of all other subsets of D_i^S, and, in addition, whose union of semi-lattices is also distinct from the union of semi-lattices of all non-empty subsets of every D_j^S with $i \neq j$. Thus, by Theorem 6.7, each of the non-empty subsets of D_i^S gives rise to a new stable CI structure. Hence, from each D_i^S, we get $2^{\binom{|S|}{2}\binom{|S|-2}{i}} - 1$ distinct stable CI structures. Since i ranges from 0 to $|S| - 2$, the statement of the lemma follows.

Example 6.16. For $|S| = 3$ there are 8 Markov networks, 22 unrestricted [**46**], and 14 stable CI structures. For $|S| = 4$ there are 64 Markov networks [**46**], 18,478 unrestricted [**47**], and at least 4,221 distinct stable CI structures. For $|S| = 5$ there are at least 2,147,485,692 distinct stable CI structures.

Using Lemma 6.15, we can show that the number of stable CI structures grows double-exponentially with the size of S.

Theorem 6.17. *The number of stable CI structures over a finite set S grows at least double-exponentially with the size of S.*

Proof: Let S be a finite set and assume without loss of generality that $|S| - 2$ is even. It is known that $\binom{n}{k} \geq (n/k)^k$ and, therefore,

$$\sum_{i=0}^{|S|-2} 2^{\binom{|S|}{2}\binom{|S|-2}{i}} \geq 2^{\binom{|S|}{2}\binom{|S|-2}{(|S|-2)/2}} \geq 2^{\binom{|S|}{2}2^{(|S|-2)/2}}.$$

Now, by Lemma 6.15, we have that the number of stable CI structures is greater than or equal to $2^{\binom{|S|}{2}2^{(|S|-2)/2}} - 1$. The proof is analogous when $|S| - 2$ is odd, where we use $\lfloor (|S| - 2)/2 \rfloor$ instead of $(|S| - 2)/2$.

4. Computational Complexity of the Stable CI Implication Problem

Recall that the stable conditional independence implication problem, denoted here as `STABLE-IMPLICATION`, is the problem of deciding, given a set of random variables S, a set of stable CI statements $\mathcal{C}$, and a CI statement c, whether $\mathcal{C}$ implies c relative to the class of discrete probability measures. In this section, we will show that `STABLE-IMPLICATION` is `coNP`-complete. Furthermore, in Section 5, we will be able to prove that a linear-time reduction exists from `STABLE-IMPLICATION` to the unsatisfiability problem, here denoted as `UNSAT`, for propositional logic formulas over variables that correspond to the random variables in S. This permits the use of `SAT` solvers to decide instances of `STABLE-IMPLICATION`. In Section 6, we present experimental results that show how such instances, even with hundreds of variables, can be decided efficiently.

First, we need to introduce the notion of minterms. Minterms are certain propositional formulas associated with subsets of a set of variables.

Definition 6.18. Let T be a set of propositional variables. Then, for each $X \subseteq T$, the *minterm* associated with X, denoted $\mathbf{X}$, is the propositional formula $\bigwedge_{a \in X} a \wedge \bigwedge_{b \in T-X} \neg b$.

Let Φ be a propositional formula over T. The *minset* of Φ, denoted *minset*(Φ), is the set $\{X \mid \mathbf{X} \models_{prop} \Phi\}$, where $\models_{prop}$ denotes the logical implication relation for propositional logic. The *negative minset* of Φ, denoted *negminset*(Φ), is the set *minset*$(\neg\Phi)$.

We will now isolate a special class of propositional formulas.

Definition 6.19. Let T be a set of propositional variables. Then `3-CNFV`(T) denotes the set of all `CNF` propositional formulas over the variables in T in which the clauses are restricted to be of the form $x \vee y$, $\neg x \vee y \vee z$, $\neg x \vee \neg y \vee z$, or $\neg x \vee \neg y \vee \neg z$, where x, y, and z are variables in T.

proposition 6.20. *Let* `3SAT-CNFV` *denote the satisfiability problem for* `3-CNFV`(T) *over sets* T *of propositional variables. Then,* `3SAT-CNFV` *is an* `NP`*-complete problem.*

Proof: Clearly, `3SAT-CNFV` is in `NP`. The hardness of `3SAT-CNFV` can be established via a reduction from the standard `3SAT` problem. Every clause in `3SAT` of the form $x \vee y \vee z$ is mapped to the formula $(x \vee y \vee \neg w) \wedge (z \vee w)$, where w is a new variable. All other clauses in `3SAT` are retained. This reduction is possible in polynomial time and preserves satisfiability.

Next, we defined a polynomial-time computable reduction from formulas in `3-CNFV` to sets of non-trivial CI statements.

Definition 6.21. Let T be a set of propositional variables and let $S = T \cup \{r, s\}$, with $r \notin T$ and $s \notin T$. Let $\mathcal{T}(S)$ denote the set of all CI statements over S. For a formula Φ in `3-CNFV`(T), let `clauses`(Φ) denote the set of clauses in Φ. Then, f : `3-CNFV`$(T) \rightarrow 2^{\mathcal{T}(S)}$ is defined as follows. For formula Φ,

$$f(\Phi) = \bigcup_{c \in \texttt{clauses}(\Phi)} f(c),$$

with[1]

$$\begin{aligned}
f(x) &= \{I(x, v|\emptyset) \mid v \in S - \{x\}\}; \\
f(\neg x) &= \{I(u, w|x) \mid u, w \in S - \{x\}, u \neq w\}; \\
f(x \vee y) &= \{I(x, y|\emptyset)\}; \\
f(\neg x \vee y) &= \{I(y, v|x) \mid v \in S - \{x, y\}\}; \\
f(\neg x \vee \neg y) &= \{I(v, w|xy) \mid v, w \in S - \{x, y\}, v \neq w\} \\
f(\neg x \vee y \vee z) &= \{I(y, z|x)\}; \\
f(\neg x \vee \neg y \vee z) &= \{I(z, v|xy) \mid v \in S - \{x, y, z\}\}; \\
f(\neg x \vee \neg y \vee \neg z) &= \{I(v, w|xyz) \mid v, w \in S - \{x, y, z\}, v \neq w\}.
\end{aligned}$$

Notice that the mapping f can be computed in polynomial time, and that, for each formula Φ, for each clause $\mathbf{c} \in$ `clauses`(Φ), and for each $X \subseteq T$, we have that $X \in \mathcal{L}(f(c))$ if and only if $\mathbf{X} \models_{prop} \neg \mathbf{c}$.

[1]To simplify the mapping, we assume that every formula in `3-CNFV`(T) is preprocessed to transform clauses with duplicate literals (e.g., $\ell \vee \ell$ or $\neg\ell \vee \neg\ell \vee \neg\ell$) into their simplified forms (here: ℓ and $\neg\ell$). Of course, this preprocessing step is computable in polynomial time.

EXAMPLE 6.22. Let $T = \{a, b, c\}$, let $S = T \cup \{d, e\}$, and let $\Phi = (a \vee c) \wedge (\neg a \vee \neg b \vee c)$. Then

$$\begin{aligned} f(\Phi) &= f(a \vee c) \cup f(\neg a \vee \neg b \vee c) \\ &= \{I(a, c|\emptyset)\} \cup \{I(c, d|ab), I(c, e|ab)\} \\ &= \{I(a, c|\emptyset), I(c, d|ab), I(c, e|ab)\}. \end{aligned}$$

Furthermore,

$$\begin{aligned} \mathcal{L}(f(\Phi)) &= \{\emptyset, b, d, e, bd, be, de, bde, ab, abe, abd\}, \text{ and} \\ \mathit{negminset}(\Phi) &= \{\mathbf{X} \mid X = \emptyset \vee X = \{b\} \vee X = ab\} = \{\emptyset, b, ab\}. \end{aligned}$$

We can now state the main result of this section.

THEOREM 6.23. `STABLE-IMPLICATION` *is* `coNP`*-complete.*

Proof: We first show that the co-problem of `STABLE-IMPLICATION` is in `NP`. Let $\mathcal{C}$ be a set of stable CI statements over S and let c be a CI statement over S. Since, by Theorem 6.7, $\mathcal{C} \not\models c$ if and only if $\mathcal{L}(\mathcal{C}) \not\supseteq \mathcal{L}(c)$, it is sufficient to guess $X \in \mathcal{L}(c) - \mathcal{L}(\mathcal{C})$ and then verify in polynomial time that, for all $I(A, B|C) \in \mathcal{C}$, one has that $(X \supseteq A) \vee (X \supseteq B) \vee \ (X \not\supseteq C)$.

To show the hardness of `STABLE-IMPLICATION` we use a reduction from `3SAT-CNFV`. Let T be a set of propositional variables, let $S = T \cup \{r, s\}$ with $r \notin T$, $s \notin T$, let f be the function from Definition 6.21, and let $\Phi \in$ `3SAT-CNFV`(T). Then we have the following:

(1) $\mathit{negminset}(\Phi) \subseteq \mathcal{L}(f(\Phi))$; and

(2) Φ is a contradiction if and only if $\mathcal{L}(I(r, s|\emptyset)) \subseteq \mathcal{L}(f(\Phi))$.

To show (1), let $X \in \mathit{negminset}(\Phi)$. Then, there exists a clause $\mathbf{c}$ in `clauses`(Φ) such that $\mathbf{X} \models_{prop} \neg\mathbf{c}$. But then, there exists $I(x, y|U) \in f(\mathbf{c})$ such that $X \supseteq U$, $x \notin X$ and $y \notin X$ because otherwise $\mathbf{X} \models_{prop} \mathbf{c}$. It follows that $X \in \mathcal{L}(f(c))$ and, therefore, $X \in \mathcal{L}(f(\Phi))$. To show (2), let Φ be a contradiction. Notice that Φ is a contradiction if and only if $\mathit{negminset}(\Phi) = 2^T$. Now, $\mathcal{L}(I(r, s|\emptyset)) = 2^T = \mathit{negminset}(\Phi) \subseteq \mathcal{L}(f(\Phi))$, where the last inclusion follows from (1). But then, by Theorem 6.7, Φ is a contradiction if and only if $f(\Phi) \models I(r, s|\emptyset)$. Since f is computable in polynomial time, the result follows.

Property of CI	**Un-restricted**	**Stable**	**Saturated**	**Markov models**
Finite, complete axiomatization	No [**45**]	Yes	Yes [**15**]	Yes [**31**]
Implication problem	?	coNP-com.	P [**15**]	?
Perfect models	Yes [**31**]	Yes	Yes [**15**]	Yes [**31**]
Perfect binary models	No [**15**]	No	?	?

FIGURE 6.5. Summary of properties of conditional independence.

The logical and algorithmic properties of unrestricted CI, stable CI, saturated CI, and Markov models are summarized in Figure 6.5.

5. Implication Testing Using Satisfiability Solvers

In this section, we show that every set of CI statements can be reduced to a propositional formula in linear time. This, together with the results from the previous section, allows us to employ SAT solvers to decide `STABLE-IMPLICATION`. Furthermore, we will show experimentally that numerous instances of the stable CI implication problem can be decided efficiently, even if several hundreds of random variables are involved.

DEFINITION 6.24. Let $\mathcal{C}$ be a set of CI statements over S, and let $\texttt{proposition}(S)$ be the set of propositional formulas over variables in S. Let $\mathcal{T}(S)$ denotes the set of all CI statements over S. The mapping $g : 2^{\mathcal{T}(S)} \rightarrow \texttt{proposition}(S)$ is defined by $g(\mathcal{C}) = \bigwedge_{c \in \mathcal{C}} g(c)$, and $g(I(A,B|C)) = \bigwedge_{a \in A} a \vee \bigwedge_{b \in B} b \vee \bigvee_{c \in C} \neg c$, for each CI statement $I(A,B|C)$ in $\mathcal{C}$.

The mapping g can be computed in linear time in the size of $\mathcal{C}$. Now, using this mapping, we can state the following theorem.

THEOREM 6.25. *Let $\mathcal{C}$ be a set of stable CI statements over S and let c be a CI statement over S. Then $\mathcal{C} \models c$ if and only if $g(\mathcal{C}) \models_{proposition} g(c)$.*

Proof: We will again use the concepts *minset* and *negminset* introduced in Definition 6.18. Let $\mathcal{C}$ be a set of CI statements and let c be a CI statement. One can verify that $\mathcal{L}(\mathcal{C}) = \mathit{negminset}(g(\mathcal{C}))$ and $\mathcal{L}(c) = \mathit{negminset}(g(c))$. By Theorem 6.7, we have that $\mathcal{C} \models c$ if and only if $\mathcal{L}(\mathcal{C}) \supseteq \mathcal{L}(c)$. Now, the statement of the theorem follows.

Example 6.26. Let $S = \{a, b, d, e\}$, let $\mathcal{C} = \{I(a, b|\emptyset), I(d, e|a), I(d, e|b)\}$, and let $c = I(d, e|\emptyset)$. We have $g(\mathcal{C}) = (a \vee b) \wedge (d \vee e \vee \neg a) \wedge (d \vee e \vee \neg b)$ and $g(c) = d \vee e$. We also have $g(\mathcal{C}) \models_{proposition} g(c)$ if and only if $g(\mathcal{C}) \wedge \neg g(c)$ is not satisfiable. Now, $g(\mathcal{C}) \wedge \neg g(c) = (a \vee b) \wedge (d \vee e \vee \neg a) \wedge (d \vee e \vee \neg b) \wedge \neg d \wedge \neg e$. This formula is a contradiction. Hence, $\mathcal{C} \models c$ by Theorem 6.25.

5.1. Concise Representation of Stable CI Structures. In this section, we study the notion of an *irredundant equivalent subset* of a set of stable CI statements. We will use this notion to represent a stable CI structure more concisely. For this purpose, we will adopt terminology which was recently introduced in the context of redundancy of propositional formulas in conjunctive normal form [**22**].

Definition 6.27. A set of CI statements $\mathcal{C}$ over S is *irredundant* if $\mathcal{C} - \{c\} \not\models c$, for all $c \in \mathcal{C}$. Otherwise, it is *redundant*.

A related definition is that of an irredundant equivalent subset. Note that a set of stable CI statements may have several different irredundant equivalent subsets and that the cardinality of these sets can differ.

Definition 6.28. Let $\mathcal{C}$ be a set of stable CI statements over S. A set of stable CI statements $\mathcal{C}'$ is an *irredundant equivalent subset* of $\mathcal{C}$ if and only if

(1) $\mathcal{C}' \subseteq \mathcal{C}$;
(2) $\mathcal{C}' \models c$ for all $c \in \mathcal{C}$; and
(3) $\mathcal{C}'$ is irredundant.

Example 6.29. Let $S = \{a, b, c\}$ and let $\mathcal{C} = \{I(a, b|\emptyset), I(a, b|c)\}$. Then, $\mathcal{C}' = \{I(a, b|\emptyset)\}$ is an irredundant equivalent subset of $\mathcal{C}$.

variables	50	100	200	300	400
time [ms]	740	1523	3362	5627	7076

FIGURE 6.6. Average time (in milliseconds) needed to decide the implication problem for different numbers of variables and 100,000 antecedents.

We now have the following property.

PROPOSITION 6.30. *Let $\mathcal{C}$ be a set of CI statements over S. Then $\mathcal{C}$ is irredundant if and only if, for all c in $\mathcal{C}$, we have that $g(\mathcal{C} - \{c\}) \wedge \neg g(c)$ is satisfiable, where g is the mapping defined in Definition 6.24.*

6. Experiments

In a first experiment, we randomly generated instances of the stable CI implication problem with up to 400 variables. We then used the mapping g from Definition 6.24 to transform instances of the implication problem for stable CI into instances of the unsatisfiability problem of propositional logic (UNSAT), to which we applied a SAT solver. We used MiniSat[2] by Niklas Eén and Niklas Sörensson on a Pentium4 dual-core Linux system for the experiments. The performance of the SAT solver is quite remarkable. Figure 6.6 shows the average time (out of 10 tests) needed to decide the implication problem $\mathcal{C} \models c$ for $|\mathcal{C}| = 100,000$ and different numbers of variables.

The goal of the second experiment was to determine the average size of irredundant equivalent subsets of a randomly generated set of stable CI statements. The algorithm in Figure 6.7 is based on Corollary 6.30. It takes as input a set of stable CI statements $\mathcal{C}$ and returns an irredundant equivalent subset of $\mathcal{C}$ based on several satisfiability tests. For each number of variables from 5 to 25, we randomly created sets of 500 CI statements and determined the size of the irredundant equivalent subsets using the algorithm. Figure 6.8 shows the average size of 1000 different runs. As one can expect, the fewer variables there are, the smaller is the irredundant equivalent subset. For the 500 satisfiability tests made

[2] http://minisat.se

irredundant-subset ($\mathcal{C}$: set) $\mathcal{C}'$: set

$\mathcal{C}' := \mathcal{C}$
for each $c \in \mathcal{C}'$
  begin
    if $g(\mathcal{C}' - \{c\}) \wedge \neg g(c)$ *not* satisfiable
      then $\mathcal{C}' := \mathcal{C}' - \{c\}$
  end
return $\mathcal{C}'$

FIGURE 6.7. A function to compute an irredundant equivalent subset.

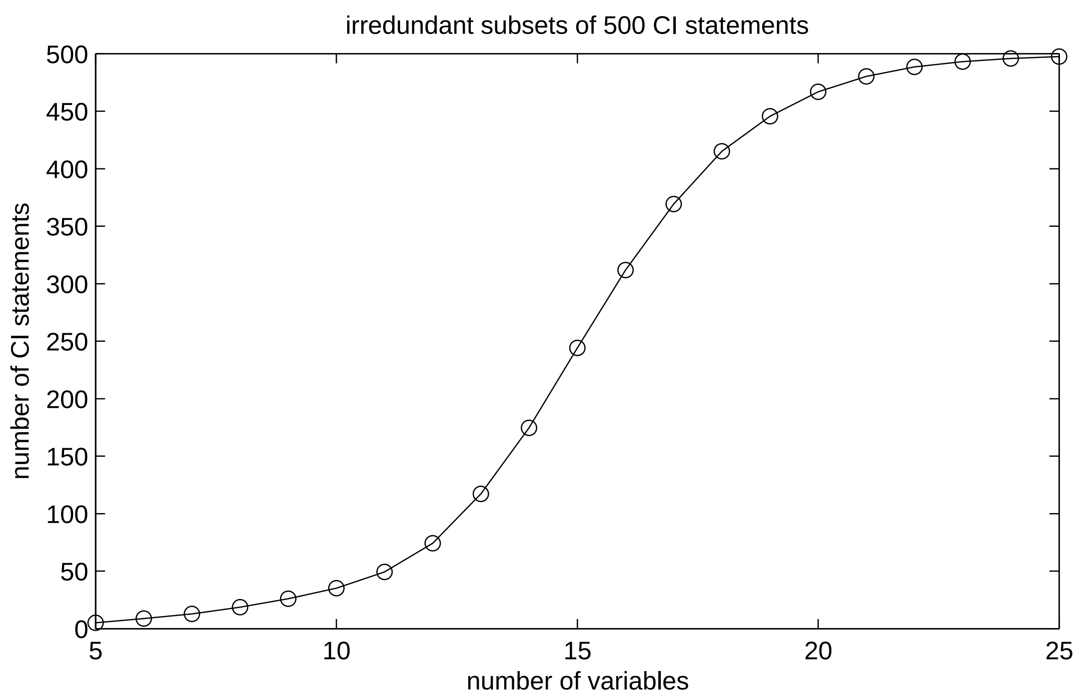

FIGURE 6.8. Size of irredundant equivalent subset of a set of initially 500 stable CI statements for different numbers of attributes.

to compute an irredundant equivalent subset, the algorithm took at most 1100 ms, where the majority of the time was spent on unsatisfiable instances of the problem. This amounts on average to 2ms per satisfiability test for sets of 500 CI statements.

CHAPTER 7

Conclusion

1. Summary

The thesis describes and develops a theoretical framework that unifies several important implication problems occurring in areas of computer science such as database systems, uncertainty in artificial intelligence, and game theory. We demonstrated the syntactic and semantic commonalities of numerous seemingly different instances of these implication problems. We first introduced the notion of a *disjunctive statement with bounded order* to achieve syntactic uniformity. Secondly, we showed that, in many instances, disjunctive statements can be interpreted as additive constraint on specific classes of real-valued functions. We then considered inference systems which are sound and/or complete for several well-known implication problems: inference system $\mathcal{K}$ and a generalization of the *semi-graphoid* axioms for statements of arbitrary order that we referred to as system $\mathcal{G}$. These inference systems serve as *templates* for specialized inference systems in the context of implication problems with bounds on the order of the disjunctive statements. We proved these inference systems to be sound and complete with respect to semi-lattice inclusion, that is, if the semi-lattice of a disjunctive statement is a subset of the union of semi-lattices of a set of CI statements, then the single CI statement can be derived from the set of CI statements. We leveraged this characterization of the inference systems to derive testable properties that imply the soundness and/or completeness of the two inference systems. Furthermore, we specified properties of real-valued functions that imply the non-existence of a finite axiomatization for the implication problem for disjunctive statements interpreted as additive constraints on specific classes of real-valued functions. These properties hold for many classes of real-valued functions and allowed us to solve some open problems in several areas of computer science. The theory builds on, and generalizes previous ideas of, non-existence proofs in the areas of embedded multivalued dependencies [**35**] and conditional independence in the area of reasoning under uncertainty [**44**]. In doing so, we achieved our goal of developing a general theoretical framework applicable to a variety of instances of the implication problem for disjunctive statements.

In Chapter 5, we applied the framework to the area of reasoning under uncertainty. One of our main contributions here was to extend the semi-graphoids to a *finite* inference system, denoted by $\mathcal{A}$, which we showed to be (1) sound and complete for saturated CI statements, (2) complete for general CI statements, and (3) sound and complete for stable CI statements (de Waal and van der Gaag [**9**]), all relative to the class of discrete probability measures. To make the connection between the theoretical framework for disjunctive statements and the conditional independence implication problem, we harnessed the concept of *multi-information functions* induced by probability measures (Studený [**46**]). This class of real-valued functions allowed us to link the implication problem for additive constraints on multi-information functions to the probabilistic CI implication problem. The combination of lattice-inclusion techniques and the completeness result for conditional independence statements made it possible to derive further properties that can be used to falsify instances of the implication problem. Additionally, we introduced a novel validation algorithm that leverages the falsification algorithm and represents implication problems as instances of linear programming problems. We demonstrated experimentally that the falsification and validation criteria, some of which can be tested in polynomial time, work very effectively. Where possible, the experimental results were related to those obtained from a *racing algorithm* introduced by Bouckaert and Studený [**3**].

Finally, in Chapter 6, we harnessed the theoretical framework to further investigate the logical and algorithmic properties of stable conditional independence structures. While the notion of stable CI existed before, this work studied it from a different angle, namely as an alternative to graphical models in representing and reasoning with conditional independence. We showed that each stable CI structure has a perfect model, i.e., a discrete probability measure that satisfies precisely the CI statements in the structure, and that the number of instances that can be modelled using stable CI grows at least double-exponentially with the number of random variables. This proved that stable CI structures are a powerful generalization of Markov networks, which can model only a small subset of all stable CI structures. Furthermore, the theory established a direct connection between sets of stable CI statements and propositional formulae in conjunctive normal form, a connection which was leveraged to show that the implication problem for stable conditional independence is coNP-complete. After deriving a linear time reduction to the Boolean satisfiability problem (SAT), we could verify empirically that existing SAT solvers can be employed to efficiently decide the implication problem and to compute concise, non-redundant representations of stable CI structures, even for instances involving hundreds of variables. Hence, stable conditional independence was shown to be a viable alternative for representing and reasoning with conditional independence information, one that is more general than graphical models but still computationally feasible.

2. Future Work

We intend to apply the theory of disjunctive statements to different calculi in artificial intelligence (e.g., possibility theory), alternative notions of conditional independence, and constraints on more complex structures such as trees. Indeed, several important methods in machine learning and game theory can be understood as instances of optimization problems on submodular or supermodular functions. Remember that additive constraint can be expressed in form of differentials. The first and second-order additive constraints are first and second-order differentials set to zero, or, equivalently, first and second order critical points of the function under consideration. The implication problem on disjunctive statements interpreted as additive constraints can now be understood as the problem of determining all the critical points that are logically implied by a set of given ones, relative to a specific class of real-valued functions. We will also carry forward research on alternative paradigms to represent and model conditional independence information. An interesting question in this regard is how we can use the concept of stable CI to learn the structure of Markov networks, and to develop measures for the degree of "faithfulness" of these networks.

There are several open problems, both theoretical and practical, in the area of reasoning under uncertainty which we will continue to work on. For example, the decidability of the implication problem, the succinct representation of CI statements, logical properties of specific CI models, and novel approaches to probabilistic inference. The theory of imsets (Studený [**46**]) provides an algebraic characterization of supermodular functions, and we believe that insightful relationships between the two theories can be established.

In this thesis, we considered implication problems in which the satisfaction of a disjunctive statement $X \rightarrow \{Y_1, \dots, Y_n\}$, relative to a real-valued function F, is determined by checking whether the additive constraint that corresponds to the disjunctive statement holds. The notion of the implication problems considered and the concept of satisfaction can be generalized as follows. Let $X \rightarrow \{Y_1, \dots, Y_n\}$ be a disjunctive statement, and let a and b be real numbers, and let F be a real-valued function. We say that F satisfies the *approximate* disjunctive statement $(X \rightarrow \{Y_1, \dots, Y_n\}, a, b)$ (abbreviated, $F \models (X \rightarrow \{Y_1, \dots, Y_n\}, a, b))$ if $\Delta_F^{\{Y_1,\dots,Y_n\}}(X) \in [a, b]$. (Observe hat the notion of satisfaction used in this thesis concides with the special case where $a = b = 0$.) Based on this notion of satisfaction and the more general class of approximate disjunctive statement, we can now consider the implication for such approximate disjunctive statements relative to certain classes of real-valued functions.

It would be interesting to see if the techniques developed in this thesis can be applied to this more general class of implication problems. In this regard, we conjecture that the results of Calders [**4**] will be useful in the context of satisfiability and implication problems for *frequency constraints* in the domain of frequent-itemset mining. In addition, there exists a substantial literature on *approximate functional dependencies* [**19, 12**] in the context of relational databases. In our framework, functional dependencies are just a special case of disjunctive statement interpreted as additive constraints on the class of Shannon entropy functions.

Probability measures are just a special class of real-valued functions that are used in the area of "reasoning under uncertainty." Other such functions are *Dempster-Shafer belief and plausibility functions*, *possibility measures*, *ranking functions*, *fuzzy measures*, and *plausibility measures* (see Halpern's book *Reasoning about Uncertainty* [**17**] for an excellent treatment of these functions.) It would be interesting to investigate how our techniques apply to these classes of uncertainty measures. In this regard, we wish to point out work by Sayrafi et al. [**40**] that undertook work in this direction for disjunctive statement or order 1 and 2. Furthermore, as synthesized in the above-mentioned book by Halpern, other notions of conditional independence have been introduced relative to the classes of uncertainty measures discussed above. It is natural to consider how our results on the CI implication problem for the class of probability measures applies, or generalizes, to these other forms of conditional independence.

This thesis has developed a method to unify a variety of implication problems occurring in various areas in computer science. This unification now also permits *cross-fertilization* between these areas. For example, functional and multivalued dependency implication problems have led to a sophisticated *decomposition theory* in relational databases [**14, 24**]. It is natural to investigate how this decomposition theory can be carried over to other computer science areas and applications. As we have seen in Chapters 5 and 6, multivalued dependencies have a close connection to saturated and stable CI statements. Therefore, any technique and result in relational database decomposition theory which makes exclusive use of the implication problem for multivalued dependencies has a direct analogue for applications where CI implication can be used. Whether these analogues are natural or useful is not clear, but it is worth considering if this is the case or not. In this regard, we also think that our results can be used in the area of frequent itemset data mining because of the close, though not entirely perfect, correspondence of functional dependencies and association rules, and multi-valued dependencies and disjunctive rules. Finally, there are many other existing implication problems in the literature. For example, in propositional logic, there is the classic logical implication problem, where one is given a set of propositional formulas $\phi_1, \ldots, \phi_n$ and another propositional formula ϕ and one is asked if $(\phi_1, \ldots, \phi_n)$ logically implies ϕ. Of course, this implication problem can be translated into a set of equivalent implication problems for disjunctive statements since each propositional formula is logically equivalent to a set of disjunctive propositional statements. We anticipate that there are other implication problems in computer science which can be translated into equivalent implication problems for disjunctive statements. Obviously, our theory should therefore be useful.

Bibliography

[1] R. A. Ash. *Information Theory*. Dover Publications, Inc. New York, 1965.

[2] C. Beeri, R. Fagin, and J. H. Howard. A complete axiomatization for functional and multivalued dependencies in database relations. In *Proceedings of the 1977 ACM SIGMOD international conference on Management of data*, pages 47–61. ACM Press, 1977.

[3] R. R. Bouckaert and M. Studený. Racing algorithms for conditional independence inference. *Int. J. Approx. Reasoning*, 45(2):386–401, 2007.

[4] T. Calders. *Axiomatization and Deduction Rules for the Frequency of Itemsets*. 2003.

[5] A. Ceglar and J. F. Roddick. Association mining. *ACM Comput. Surv.*, 38(2), 2006.

[6] G. Choquet. Theory of capacities. *Annales de l'institut Fourier*, 5:131–295, 1954.

[7] M. M. Dalkilic and E. L. Robertson. Information dependencies. In *Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 245–253. ACM Press, 2000.

[8] A. P. Dawid. Conditional independence in statistical theory. *Journal of the Royal Statistical Society*, 41:1–31, 1979.

[9] P. de Waal and L. C. van der Gaag. Stable independence and complexity of representation. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pages 112–119, 2004.

[10] P. R. de Waal and L. C. van der Gaag. Stable independence in perfect maps. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*, pages 161–168, 2005.

[11] J. Demetrovics, L. Libkin, and I. B. Muchnik. Functional dependencies in relational databases: A lattice point of view. *Discrete Applied Mathematics*, 40(2):155–185, 1992.

[12] J. T. Engle and E. L. Robertson. Hls: Tunable mining of approximate functional dependencies. In *BNCOD*, pages 28–39, 2008.

[13] P. Gandhi, F. Bromberg, and D. Margaritis. Learning markov network structure using few independence tests. In *Proceedings of the SIAM International Conference on Data Mining*, pages 680–691, 2008.

[14] H. Garcia-Molina, J. D. Ullman, and J. Widom. *Database Systems: The Complete Book*. Prentice-Hall, 2008.

[15] D. Geiger and J. Pearl. Logical and algorithmic properties of conditional independence and graphical models. *The Annals of Statistics*, 21(4):2001–2021, 1993.

[16] S. Goldberg. *Introduction to Difference Equations*. Dover, 1986.

[17] J. Y. Halpern. *Reasoning about Uncertainty*. The MIT Press, 2003.

[18] R. Hemmecke, J. Morton, A. Shiu, and B. Sturmfels. Three counter-examples on semi-graphoids. *Combinatorics, Probability and Computing*, 17(2):239–257, 2008.

[19] Y. Huhtala, J. Kärkkäinen, P. Porkka, and H. Toivonen. Tane: An efficient algorithm for discovering functional and approximate dependencies. *The Computer Journal*, 42(2):100–111, 1999.

[20] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

[21] T. T. Lee. An information-theoretic analysis of relational databases – part i: Data dependencies and information metric. *IEEE Transactions on Software Engineering*, SE-13:1049–1061, 1987.

[22] P. Liberatore. Redundancy in logic i: Cnf propositional formulae. *Artif. Intell.*, 163(2):203–232, 2005.

[23] R. D. Luce and H. Raiffa. *Games and Decisions, Introduction and Critical Survey*. John Wiley and Sons, Inc., 1957.

[24] D. Maier. *The Theory of Relational Databases*. Computer Science Press, 1983.

[25] D. Maier. *The Theory of Relational Databases*. Computer Science Press, Inc., 1988.

[26] F. M. Malvestuto. Statistical treatment of the information content of a database. *Information Systems*, 11:211–223, 1986.

[27] F. Matúš. Ascending and descending conditional independence relations. In *Transactions of the 11th Prague Conference on Information Theory*, pages 189–200, 1992.

[28] M. Niepert. Logical inference algorithms and matrix representations for probabilistic conditional independence. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, 2009.

[29] M. Niepert and D. Van Gucht. Logical properties of stable conditional independence. In *Proceedings of the 4th European Workshop on Probabilistic Graphical Models*, pages 225–232, 2008.

[30] M. Niepert, D. Van Gucht, and M. Gyssens. On the conditional independence implication problem: A lattice-theoretic approach. In *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*, pages 435–443, 2008.

[31] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., 1988.

[32] J. Pearl and A. Paz. Graphoids: Graph-based logic for reasoning about relevance relations or when would x tell you more about y if you already know z? In *Advances in Artificial Intelligence II, Seventh European Conference on Artificial Intelligence*, pages 357–363, 1986.

[33] J. Pearl and A. Paz. Graphoids, graph-based logic for reasoning about relevance relations. *Advances in Artificial Intelligence II*, pages 357–363, 1987.

[34] J. Peña, R. Nilsson, J. Björkegren, and J. Tegnér. Reading dependencies from the minimal undirected independence map of a graphoid that satisfies weak transitivity. In *Proccedings of the European Workshop on Probabilistic Graphical Models*, pages 247–254, 2006.

[35] Y. Sagiv and S. F. Walecka. Subset dependencies and a completeness result for a subclass of embedded multivalued dependencies. *Journal of the ACM*, 29(1):103–117, 1982.

[36] B. Sayrafi. *A Measure-Theoretic Framework For Constraints and Bounds on Measurements of Data*. PhD dissertation- Indiana University, 2005.

[37] B. Sayrafi and D. Van Gucht. Inference systems derived from additive measures. *Workshop on Causality and Causal Discovery, London, Canada*, 2004.

[38] B. Sayrafi and D. Van Gucht. Differential constraints. In *Proceedings of the ACM Symposium on Principles of Database Systems*, pages 348–357, 2005.

[39] B. Sayrafi, D. Van Gucht, and M. Gyssens. Measures in databases and data mining. Tech. Report TR602, Indiana University Computer Science, 2004.

[40] B. Sayrafi, D. Van Gucht, and M. Gyssens. The implication problem for measure constraints. *Information Systems*, 2008.

[41] B. Sayrafi, D. Van Gucht, and P. W. Purdom. On the effectiveness and efficiency of computing bounds on the support of item-sets in the frequent item-sets mining problem. In *OSDM '05: Proceedings of the 1st international workshop on open source data mining*, pages 46–55, 2005.

[42] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, 1986.

[43] M. Studený. Multiinformation and the problem of characterization of conditional independence relations. *Problems of Control and Information Theory*, 18(1):3–16, 1989.

[44] M. Studený. Conditional independence statements have no complete characterization. In *Transactions of the 11th Prague Conference on Information Theory*, pages 377–396, 1992.

[45] M. Studený. Conditional independence statements have no complete characterization. In *Proceedings of the 11th Prague Conference on Information Theory*, pages 377–396, 1993.

[46] M. Studený. *Probabilistic Conditional Independence Structures*. Springer-Verlag, 2005.

[47] P. Šimeček. A short note on discrete representability of independence models. In *Proccedings of the European Workshop on Probabilistic Graphical Models*, pages 287–292, 2006.

Curriculum Vitae

Department of Computer Science
Lindley Hall 215
150 S. Woodlawn Ave.
Indiana University
Bloomington, IN 47405-7104

Phone: (812) 369-1345
Fax: (812) 855-4829
`mniepert@indiana.edu`
`http://research.matlog.net`

Education

Ph.D. Computer Science, Indiana University Bloomington, USA, 2009
Minors: Mathematics and Artificial Intelligence
Advisor: Professor Dirk Van Gucht

Graduate Studies in Computer and Cognitive Science, University of Freiburg, Germany, 2002–2004

Vordiplom (B.S. equiv.) Computer Science, Clausthal University of Technology, Germany, 2002
("Sehr Gut", best possible grade)

Research Interests

Artificial Intelligence, Data Mining, Machine Learning, Reasoning under Uncertainty, Formal Ontologies, Ontology Learning and Population, Answer Set Programming, Database Theory.

Research and Professional Experience

Visitor — *September–October 2008*
Professor Marc Gyssens — Hasselt University, Belgium
Visit of the Theoretical Computer Science Group at Hasselt University & Transnational University of Limburg in Belgium to work on my research project with Professor Marc Gyssens.

Visitor — *July 2008*
Dr. Edward N. Zalta — CSLI, Stanford University
Worked on the Indiana Philosophy Ontology project and the introduction of some of its components to the Stanford Encyclopedia of Philosophy management system. Implemented cross-referencing engine and developed the InPhO taxonomy search interface.

Research Assistant — *2006–2008*
Professor Colin Allen — Cognitive Science Program, Indiana University
Developed the Indiana Philosophy Ontology project in collaboration with philosophers and cognitive scientists, had responsibility for most computer science related aspects of the system, wrote research papers, and participated in writing grant applications for funding agencies NEH and NSF.

Graduate Student Researcher — *2005–2008*
Professor Dirk Van Gucht — Dept. of Computer Science, Indiana University
Under the supervision of Professor Dirk Van Gucht, I developed a theoretical framework that unifies several implication problems occurring in data mining, uncertain reasoning, and game theory. It involves methods from mathematical logic, probability theory, and lattice theory.

Software Developer — *Summer 2007*
Professor Katy Börner — Cyberinfrastructure for Network Science Center
Designed and implemented user interfaces and databases for a novel lab management system with visualization capabilities at the Cyberinfrastructure for Network Science Center, Indiana University, under the supervision of Associate Professor Katy Börner.

Undergraduate Research Assistant *2003–2004*

Fraunhofer Institute for Solid-State Physics Freiburg, Germany

Programmed lab equipment for experiments, employed AI algorithms to optimize the mounting of optical fibers to network hardware components, and gathered, stored, and explored experiment data in collaboration with a team of physicists.

Software Developer *2000–2008*

Friedrichstift Leimen Heidelberg, Germany

I had the responsibility for the IT infrastructure of the institution with 40 employees. Installed and maintained the intranet and developed software for the company's website and its management.

Emergency Medical Technician *1999–2000*

Red Cross Heidelberg Heidelberg, Germany

Teaching Experience

Associate Instructor, Theory of Computation, Indiana University Fall 2005

Associate Instructor, Algorithm Design and Analysis, Indiana University Spring 2006

Associate Instructor, Advanced Database Concepts, Indiana University Fall 2007

Associate Instructor, Discrete Mathematics for Computer Science, Indiana University Spring 2008

Guest Lecturer for classes *Computer Structures* and *Data Mining*, Indiana University 2008

Selected Publications

Refereed Conference & Workshop Publications

7. Mathias Niepert. **Logical Inference Algorithms and Matrix Representations for Probabilistic Conditional Independence.** In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, Montreal, Canada, 2009.

6. Mathias Niepert, Cameron Buckner, and Colin Allen. **Working the Crowd: Design Principles and Early Lessons from the Social-Semantic Web.** In *Proceedings of the Workshop on Web 3.0: Merging Semantic Web and Social Web - (SW)*^{2} at ACM Hypertext, Turin, Italy, 2009.

5. Mathias Niepert and Dirk Van Gucht. **Logical Properties of Stable Conditional Independence.** In *Proceedings of the 4th European Workshop on Probabilistic Graphical Models*, Hirtshals, Denmark, pages 225–232, 2008. (selected for plenary presentation)

4. Mathias Niepert, Dirk Van Gucht, and Marc Gyssens. **On the Conditional Independence Implication Problem: A Lattice-Theoretic Approach.** In *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence (UAI)*, Helsinki, Finland, pages 435–443, AUAI Press, 2008. (**best student paper runner-up award**; extended version will be submitted to the journal *Artificial Intelligence* following an invitation)

3. Mathias Niepert, Cameron Buckner, Jaimie Murdock, and Colin Allen. **InPhO: A System for Collaboratively Populating and Extending a Dynamic Ontology.** *(demo abstract)*. In *Proceedings of the 8th IEEE/ACM Joint Conference on Digital Libraries*, 2008, Pittsburgh, Pennsylvania, 429, ACM Press, 2008.

2. Mathias Niepert, Cameron Buckner, and Colin Allen. **Answer Set Programming on Expert Feedback to Populate and Extend Dynamic Ontologies.** In *Proceedings of the 21st International FLAIRS Conference*, Coconut Grove, Florida, pages 500–505, AAAI Press, 2008.

1. Mathias Niepert, Cameron Buckner, and Colin Allen. **A Dynamic Ontology for a Dynamic Reference Work.** In *Proceedings of the 7th IEEE/ACM Joint Conference on Digital Libraries*, Vancouver, British Columbia, pages 288–297, ACM Press, 2007.

Refereed Journal Publications

4. Mathias Niepert, Dirk Van Gucht, and Marc Gyssens. **On the Conditional Independence Implication Problem: A Lattice-Theoretic Approach.** (in preparation)

3. Mathias Niepert, Dirk Van Gucht, and Marc Gyssens. **Logical and Algorithmic Properties of Stable Conditional Independence.** *International Journal of Approximate Reasoning*, Elsevier, 2009. (forthcoming)

2. Cameron Buckner, Mathias Niepert, and Colin Allen. **From Encyclopedia to Ontology: Toward A Dynamic Representation of the Discipline of Philosophy.** In a special issue of *Synthese*, Springer-Verlag, 2009. (forthcoming)

1. Colin Allen, Cameron Buckner, and Mathias Niepert. **The World is Not Flat: Expertise and InPhO.** Selected papers from the Ninth Annual WebWise Conference. *First Monday*, Volume 13, Number 8, 2008.

Miscellaneous Publications

2. Mathias Niepert, Cameron Buckner, Jaimie Murdock, and Colin Allen. **InPhO: A System for Collaboratively Populating and Extending a Dynamic Ontology.** *Bulletin of the IEEE Technical Committee on Digital Libraries*, Volume 5, Issue 1, 2009.

1. Cameron Buckner, Mathias Niepert, and Colin Allen. **InPhO: The Indiana Philosophy Ontology.** *APA Newsletter on Philosophy and Computers*, Volume 7, Number 1, 26-28, 2007.

Software

1. **TCEJ – Text Classification Environment in Java.** A graphical environment for conducting text classification experiments. Users can load several different text corpora, weighting functions, dimensionality reduction policies, and classifiers. 2005. `http://tcej.matlog.net`

Awards and Honors

Best student paper runner-up award, Conference on Uncertainty in AI	2008
Indiana University Office of International Services Scholarship	2008
Travel Grant from the Association for Uncertainty in Artificial Intelligence	2008, 2009
School of Informatics Paul W. Purdom Fellowship (awarded annually to one student)	2008–2009
NEH Digital Humanities Start-Up Grant, with Colin Allen (PI)	2007–2008
Funding for Conference Travel, Computer Science Department	2008
Funding for Conference Travel, Cognitive Science Program	2007
IU New Frontiers in the Arts and Humanities Grant, with Colin Allen (PI)	2006–2007
Indiana University Graduate Fellowship	2004–2005

Conference Presentations & Invited Talks

On Implication Problems for Disjunctive Constraints. Logic Seminar, Indiana University, February 11th, 2009.

Logical Properties of Stable Conditional Independence. Data and Search Institute Seminar, Indiana University, November 19th, 2008.

Logical Properties of Stable Conditional Independence. Workshop on Probabilistic Graphical Models, Hirtshals, Denmark, September 17th, 2008.

On the Conditional Independence Implication Problem: A Lattice-Theoretic Approach. Conference on Uncertainty in Artificial Intelligence, Helsinki, Finland, July 12th, 2008.

Answer Set Programming on Expert Feedback to Populate and Extend Dynamic Ontologies. International FLAIRS Conference, Coconut Grove, Florida, May 17th, 2008.

The Implication Problem for Disjunctive Statements. Research Seminar in Databases and Data Mining, Indiana University, September 24, 2007.

A Dynamic Ontology for a Dynamic Reference Work. *North American Computers and Philosophy (NA-CAP) Conference*, Loyola University, Chicago, July 28, 2007.

A Dynamic Ontology for a Dynamic Reference Work. *IEEE/ACM Joint Conference on Digital Libraries*, Vancouver, Canada, June 22, 2007.

Combining Statistical Language Processing and Defeasible Reasoning for Collaborative Ontology Learning. *Research Seminar in Databases and Datamining*, Indiana University, March 5, 2007.

Ontology Learning and Population from Text. *Research Seminar in Databases and Datamining*, Indiana University, February 13, 2006.

Professional Service and Associations

Member of the Association for Computing Machinery	2007–2009
PC, International Semantic Web Conference (ISWC) Reasearch Track	2009
Reviewer, Symposium on Principles of Database Systems (PODS)	2009
Reviewer, Foundations of Information and Knowledge Systems (FoIKS)	2008
Conference Volunteer, ACM/IEEE Joint Conference on Digital Libraries	2008
Moderator and organizer, Informatics/Computer Science Graduate Poster Session	2008
Volunteer Tutor, University of Freiburg, Germany	2003, 2004
Volunteer Tutor, Clausthal University of Technology, Germany *(Volunteered to tutor international students in Mathematics and Computer Science)*	2001, 2002