ON VARIANTS OF DEPENDENCE LOGIC

AXIOMATIZABILITY AND EXPRESSIVENESS

MIIKA HANNULA

ACADEMIC DISSERTATION

To be presented, with the permission of the Faculty of Science of the University of Helsinki, for public examination in Auditorium XII, University main building (Unioninkatu 34), on 29 June 2015, at 12 o'clock noon.

Department of Mathematics and Statistics
Faculty of Science
University of Helsinki
2015

Supervisor
Docent Juha Kontinen, University of Helsinki, Finland.

Pre-examiners
Professor Martin Otto, TU Darmstadt, Germany.
Professor Bernhard Thalheim, University of Kiel, Germany.

Opponent
Professor Phokion Kolaitis, University of California, Santa Cruz, USA.

Custos
Professor Jouko Väänänen, University of Helsinki, Finland.

ISBN 978-951-51-1245-3 (paperback)
ISBN 978-951-51-1246-0 (PDF)
`http://ethesis.helsinki.fi`

Printed at Unigrafia

Helsinki 2015

Abstract

Dependence logic is a novel logical formalism that has connections to database theory, statistics, linguistics, social choice theory, and physics. Its aim is to provide a systematic and mathematically rigorous tool for studying notions of dependence and independence in different areas. Recently many variants of dependence logic have been studied in the contexts of first-order, modal, and propositional logic. In this thesis we examine independence and inclusion logic that are variants of dependence logic extending first-order logic with so-called independence or inclusion atoms, respectively. The work consists of two parts in which we study either axiomatizability or expressivity hierarchies regarding these logics.

In the first part we examine whether there exist some natural parameters of independence and inclusion logic that give rise to infinite expressivity or complexity hierarchies. Two main parameters are considered. These are arity of a dependency atom and number of universal quantifiers. We show that for both logics, the notion of arity gives rise to strict expressivity hierarchies. With respect to number of universal quantifiers however, strictness or collapse of the corresponding hierarchies turns out to be relative to the choice of semantics.

In the second part we turn attention to axiomatizations. Due to their complexity, dependence and independence logic cannot have a complete recursively enumerable axiomatization. Hence, restricting attention to partial solutions, we first axiomatize all first-order consequences of independence logic sentences, thus extending an analogous result for dependence logic. We also consider the class of independence and inclusion atoms, and show that it can be axiomatized using implicit existential quantification. For relational databases this implies a sound and complete axiomatization of embedded multivalued and inclusion dependencies taken together. Lastly, we consider keys together with so-called pure independence atoms and prove both positive and negative results regarding their finite axiomatizability.

Acknowledgements

First of all, I wish to express my deepest gratitude to my supervisor Doc. Juha Kontinen for guiding and supporting me through all stages of this work. He has been a great advisor and pleasant to work with, and I have learned a lot from his optimistic and determined attitude towards research. I am also thankful for having been introduced to many different research questions and people around this fast growing field of dependence logic.

I am grateful to Prof. Jouko Väänänen for his advice during my doctoral studies (the idea for Paper IV is due to Jouko), and for acting as the custos at the doctoral defence of this dissertation. I am also thankful for the opportunity to present my work in the many dependence logic meetings that Doc. Kontinen and Prof. Väänänen have been organizing in the recent years.

I give my thanks to the whole University of Helsinki logic group for creating a stimulating atmosphere to work and study, and members of the logic group at the University of Tampere for many useful discussions, and invitations to talk at seminars. Concerning my M.Sc. studies, I wish to thank my M.Sc. thesis advisor Doc. Tapani Hyttinen for giving me my first lessons about mathematical research.

I am grateful to my co-authors, Dr. Pietro Galliani, Doc. Juha Kontinen, and Assoc. Prof. Sebastian Link, for contributing to this work. An intriguing part of my studies has been the possibility to learn from many experts in various areas around dependence logic. Besides my co-authors, and colleagues in and from Finland, I also thank Profs. Arnaud Durand, Bernhard Thalheim, and Heribert Vollmer and his group for valuable discussions.

I am thankful to the pre-examiners, Profs. Martin Otto and Bernhard Thalheim, for their careful reviews and valuable feedback. I also give my thanks to Prof. Phokion Kolaitis for kindly accepting to act as the opponent at the doctoral defence.

I sincerely acknowledge all the financial support I have received during the years of my graduate studies. I thank the Research Foundation of the University of Helsinki for supporting me at the beginning of my studies. Since 2013 I have been funded by the Academy of Finland through my advisor's research project, and I am much indebted for this opportunity to work and travel all this time without any financial concerns. I also thank the Doctoral Programme in Mathematics and Statistics for supporting a summer school journey of mine.

Finally, I wish to thank all my family and friends. In particular, I am grateful to my parents, sister, and brother for your unconditional support, my fiancée Milja for all your love, and God for making this all possible. *Kiitos.*

Helsinki, May 2015
Miika Hannula

List of Articles

This thesis consists of the introductory part and the following six articles, referred to in the text by their Roman numerals. The papers are reproduced with the permission of their respective copyright holders.

I Pietro Galliani, Miika Hannula, and Juha Kontinen. Hierarchies in independence logic. In Simona Ronchi Della Rocca, editor, *Computer Science Logic 2013 (CSL 2013)*, volume 23 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 263–280, Dagstuhl, Germany, 2013. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

II Miika Hannula and Juha Kontinen. Hierarchies in independence and inclusion logic with strict semantics. *Journal of Logic and Computation*, 2014. doi: 10.1093/logcom/exu057.

III Miika Hannula. Hierarchies in inclusion logic with lax semantics. In *Logic and Its Applications - 6th Indian Conference, ICLA 2015, Mumbai, India, January 8-10, 2015. Proceedings*, pages 100–118, 2015.

IV Miika Hannula. Axiomatizing first-order consequences in independence logic. *Annals of Pure and Applied Logic*, 166(1):61–91, 2015.

V Miika Hannula and Juha Kontinen. A finite axiomatization of conditional independence and inclusion dependencies. In Christoph Beierle and Carlo Meghini, editors, *Foundations of Information and Knowledge Systems - 8th International Symposium, FoIKS 2014, Bordeaux, France, March 3-7, 2014. Proceedings*, volume 8367 of *Lecture Notes in Computer Science*, pages 211–229. Springer, 2014.

VI Miika Hannula, Juha Kontinen, and Sebastian Link. On independence atoms and keys. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pages 1229–1238, 2014.

Author contributions. Paper I is joint work by Pietro Galliani, Juha Kontinen, and the present author. In Paper II, the characterization of $\mathrm{FO}(\subseteq)(k\forall)$-fragments was done jointly by both authors, and the prenex normal form theorem is due to the present author. The axiomatization of conditional independence and inclusion atoms in Paper V was found jointly by both authors, and the present author had a substantial part in proving its completeness. In Paper VI, the results are joint work by all authors, and the present author contributed especially to the non-axiomatizability result.

Contents

1. Overview

Dependence logic [45] ($\mathrm{FO}(=(\ldots))$) extends first-order logic (FO) with dependence atomic formulae

$$=(x_1, \ldots, x_n) \tag{1}$$

the meaning of which is that the value of x_n is functionally determined by the values of $x_1, \ldots, x_{n-1}$. In this PhD thesis we study two variants of dependence logic, independence logic ($\mathrm{FO}(\perp_c)$) and inclusion logic ($\mathrm{FO}(\subseteq)$) [24, 18]. Independence logic extends first-order logic with conditional independence atoms

$$\vec{y} \perp_{\vec{x}} \vec{z} \tag{2}$$

which express that the values of $\vec{y}$ and $\vec{z}$ are independent for any fixed value of $\vec{x}$. Inclusion logic instead extends first-order logic with inclusion atoms

$$\vec{x} \subseteq \vec{y} \tag{3}$$

which indicate that all the values of $\vec{x}$ appear also as values of $\vec{y}$. The semantics for these logics is given compositionally using sets of assignments (*teams*) instead of single assignments [31].

This thesis is divided into two subcategories. In Papers I–III we examine expressiveness hierarchies within $\mathrm{FO}(\perp_c)$ and $\mathrm{FO}(\subseteq)$. These hierarchies arise from fragments of $\mathrm{FO}(\perp_c)$ and $\mathrm{FO}(\subseteq)$ defined by restricting either the arity of non-first-order atoms or the number of universal quantifiers. In Papers IV–VI we investigate axiomatizability issues for variants of $\mathrm{FO}(=(\ldots))$.

2. Dependence and Independence

Notions of dependence and independence occur naturally in many different fields of science. Any scientific discipline dealing with large numbers of data arising from e.g. physical experiments or voting results, usually contains some study of dependencies. While this is often spesified by the context, there exist similarities between different fields. In dependence logic, one studies dependencies between variables or terms. In statistics or in database theory one examines dependencies that occur between random variables or attributes of databases. For all of these different approaches, one common research line is to examine the laws that describe the interaction between dependencies. For instance, the probabilistic independence $A \| B$ between two sets of random variables A and B has been given the following sound and complete axiomatization [22]:

- Empty set: $A \| \emptyset$,
- Symmetry: $A \| B \rightarrow B \| A$,
- Monotonicity: $A \| BC \rightarrow A \| B$,
- Exchange: $AB \| C \rightarrow (A \| B \rightarrow A \| BC)$.

Interestingly, exactly the same rules apply for the notion of independence between variables or attributes. Namely, an analogous axiomatization is sound and complete for independence atoms $\vec{x} \perp \vec{y}$, the meaning of which is that the values of $\vec{x}\vec{y}$ form a cartesian product in a team (or a relational database) [34]. While the study of dependence logic and its variants is interesting in its own right, one goal of the research is to provide new knowledge and applications that are relevant outside the dependence logic community, similarly to the above example. The novelty with dependence logic comes from the fact that different dependency notions are treated together with logical connectives and quantifiers that are each given a compositional interpretation in team semantics. This treatment of dependency notions as additions to first-order logic has potential to generate new ideas and perspectives to the study of dependencies in a variety of fields in mathematics and computer science.

3. Dependence Logic

Although introduced in 2007 by Väänänen in his book by the same name, the history of dependence logic dates back to the 1960s. First-order logic extended with branching quantifiers (Henkin 1961, [27]) is equi-expressive to dependence logic, as is independence-friendly logic (Hintikka and Sandu 1989, [30]). Henkin gives quantifiers a partial ordering, while Hintikka and Sandu introduce quantifiers of the form $\exists x/y$ (or $\forall x/y$), the meaning of which is that the choice for the value of x must not depend on the value of y. Both logics have a game theoretic semantics similar to that of first-order logic. The only exception is that the availability of information to the players can be restricted at the quantifier level. Since all the expressions in these two formalisms have equivalent second-order skolemizations, these two logics are equi-expressive to existential second order logic (ESO).

Unlike its predecessors, dependence logic does not introduce dependencies between quantifiers, but instead between variables. The syntax of dependence logic extends that of first-order logic, defined in terms of $\vee$, $\wedge$, $\neg$, $\exists$ and $\forall$, with dependence atoms of the form (1). The semantics of dependence logic is formulated using sets of assignments, called teams. For a model $\mathcal{M}$ and its domain M, an *assignment* s of M is a finite mapping from variables into M. A *team* X of M is a set of assignments of M, all sharing a common domain $\mathrm{Dom}(X)$. For an assignment s of M and $a \in M$, $s(a/x)$ denotes the assignment (with domain $\mathrm{Dom}(s) \cup \{x\}$) that agrees with s everywhere except that it maps x to a. For a team X and a mapping $F : X \to M$, we define the *supplementation team* as

$$X[F/x] := \{s(a/x) : s \in X, a = F(s)\},$$

and the *duplication team* as

$$X[M/x] := \{s(a/x) : s \in X, a \in M\}.$$

Team semantics is now given as follows. Dependence atoms are first given the following semantic rule:

- $\mathcal{M} \models_X =(x_1, \ldots, x_n)$ iff for all $s, s' \in X$ with $s(x_1) = s'(x_1), \ldots, s(x_{n-1}) = s'(x_{n-1})$, we have that $s(x_n) = s'(x_n)$.

Otherwise, we define team semantics as in Definition 4. From now on, we restrict attention to formulae where negation is only allowed to appear in front of first-order atomic formulae. We also use $\mathcal{M} \models_s \phi$ to denote the usual Tarskian semantics of first-order logic.

Definition 4 (Team Semantics)**.** For formulae ϕ, models $\mathcal{M}$ and teams X, the satisfaction relation $\mathcal{M} \models_X \phi$ is defined as follows:

- If ϕ is a first-order literal, then $\mathcal{M} \models_X \phi$ iff $\mathcal{M} \models_s \phi$ for all $s \in X$.
- $\mathcal{M} \models_X \phi \wedge \psi$ iff $\mathcal{M} \models_X \phi$ and $\mathcal{M} \models_X \psi$.
- $\mathcal{M} \models_X \phi \vee \psi$ iff there exist $Y, Z \subseteq X$, $Y \cup Z = X$ such that $\mathcal{M} \models_Y \phi$ and $\mathcal{M} \models_Z \psi$.
- $\mathcal{M} \models_X \exists x \phi$ iff there exists $F : X \to M$ such that $\mathcal{M} \models_{X[F/x]} \phi$.
- $\mathcal{M} \models_X \forall x \phi$ iff $\mathcal{M} \models_{X[M/x]} \phi$.

As an immediate consequence of the definition we obtain that the first condition generalizes to all negation normal form first-order formulae.

Proposition 5 (Flatness)**.** *Let ϕ be a first-order formula in negation normal form. Then for all models $\mathcal{M}$ and teams X:*

$$\mathcal{M} \models_X \phi \textit{ iff } \mathcal{M} \models_s \phi \textit{ for all } s \in X.$$

Next we introduce some basic properties of FO(=(...)). Unless otherwise stated, these results were proved in [45]. For a team X and $V \subseteq \mathrm{Dom}(X)$, we let $X \upharpoonright V$ to denote the team $\{s \upharpoonright V : s \in X\}$ where $s \upharpoonright V$ is the restriction of s to V. Also, let $\mathrm{Fr}(\phi)$ denote the set of free variables of ϕ, defined as for first-order logic. A formula with no free variables is called a *sentence*. The following proposition show that the truth of a formula depends only on the values of variables that occur free in the formula.

Proposition 6 (Locality)**.** *Suppose $\phi \in$ FO(=(...)) and V are such that $Fr(\phi) \subseteq V$. Then*

$$\mathcal{M} \models_X \phi \textit{ iff } \mathcal{M} \models_{X \upharpoonright V} \phi.$$

All dependence logic formulae also satisfy the following downward closure property.

Proposition 7 (Downward Closure)**.** *Let $\mathcal{M}$ be a model, X and Y teams such that $Y \subseteq X$, and $\phi \in$ FO(=(...)). Then $\mathcal{M} \models_X \phi$ implies $\mathcal{M} \models_Y \phi$.*

Over sentences, the expressive power of FO(=(...)) coincides with that of ESO. Namely, for all sentences of FO(=(...)) there exists an equivalent ESO sentence, and vice versa. Recall that ESO formulae are of the form

$$\exists R_1 \ldots \exists R_n \psi$$

where R_i is a relation or function symbol and ψ a first-order formula. For a team X with domain $\{x_1, \ldots x_n\}$, we denote by $rel(X)$ the relation

$$\{(s(x_1), \ldots, s(x_n)) : s \in X\}.$$

The following two theorems describe the expressive power of open FO(=(...)) formulae.

Theorem 8. *Let τ be a vocabulary and ϕ a* $\mathrm{FO}(=(\ldots))[\tau]$ *formula with free variables* $x_1, \ldots, x_n$*. Then there is a* $\mathrm{ESO}[\tau \cup \{R\}]$ *sentence ψ in which R appears only negatively, and such that for all models $\mathcal{M}$ and teams X with domain $\{x_1, \ldots, x_n\}$:*

$$\mathcal{M} \models_X \phi \text{ iff } (\mathcal{M}, rel(X)) \models \psi.$$

Theorem 9 ([35])**.** *Let τ be a vocabulary and $R \notin \tau$ a k-ary relation symbol. Then for every* $\mathrm{ESO}[\tau \cup \{R\}]$ *sentence ψ in which R appears only negatively, there is a* $\mathrm{FO}(=(\ldots))[\tau]$ *formula ϕ with free variables $x_1, \ldots, x_n$, and such that for all models $\mathcal{M}$ and teams $X \neq \emptyset$ with domain $\{x_1, \ldots, x_n\}$:*

$$\mathcal{M} \models_X \phi \text{ iff } (\mathcal{M}, rel(X)) \models \psi.$$

Note that these two theorems together imply that $\mathrm{FO}(=(\ldots))$ captures all downwards closed ESO definable properties of teams. Note also that the empty team $\emptyset$ satisfies all $\mathrm{FO}(=(\ldots))$ formulae, and is hence omitted in the theorem.

4. Independence and Inclusion Logic

Since the introduction of dependence logic, many variants of it have been introduced. In [24] Väänänen and Grädel defined independence logic, a variant of $\mathrm{FO}(=(\ldots))$ that extends FO with conditional independence atoms. Galliani considered adding inclusion and exclusion atoms to FO in [18]. Due to flexibility of team semantics in use, there is no *a priori* limit for extending FO with various dependency notions. Often these notions have analogous counterparts in database theory. For instance, dependence atoms correspond to functional dependencies (FDs), conditional independence atoms to embedded multivalued dependencies (EMVDs), and inclusion atoms to inclusion dependencies (INDs). Usually, since these dependency notions are first-order definable and negation is allowed only in front of first-order atoms, the expressive powers of the logics obtained do not exceed that of ESO, as is the case with $\mathrm{FO}(=(\ldots))$. However, some of the properties of $\mathrm{FO}(=(\ldots))$, such as locality or downward closure, might fail or depend on which alternative definition of team semantics is being used.

In this PhD thesis we mainly consider independence and inclusion logic which extend FO with atoms of the form (2) or (3), respectively. The semantic rules for these atoms are given as follows. For $\vec{x} := (x_1, \ldots, x_n)$. we write $s(\vec{x}) = s'(\vec{x})$ if $s(x_1) = s'(x_1), \ldots, s(x_n) = s'(x_n)$.

- $\mathcal{M} \models_X \vec{y} \perp_{\vec{x}} \vec{z}$ iff for all $s, s' \in X$ such that $s(\vec{x}) = s'(\vec{x})$, there exists $s'' \in X$ such that $s''(\vec{x}) = s(\vec{x})$, $s''(\vec{y}) = s(\vec{y})$ and $s''(\vec{z}) = s'(\vec{z})$.
- $\mathcal{M} \models_X \vec{x} \subseteq \vec{y}$ iff for all $s \in X$ there exists $s' \in X$ such that $s(\vec{x}) = s'(\vec{y})$.

The semantics of $\mathrm{FO}(\subseteq)$ or $\mathrm{FO}(\perp_c)$ is usually given as in Definition 4 except that the rule for existential quantification is a non-deterministic one. For a team X and a mapping $G : X \to \mathcal{P}(M) \setminus \{\emptyset\}$, the supplementation team is now defined as

$$X[G/x] := \{s(a/x) : s \in X, a \in G(s)\}.$$

Then *lax semantics* is obtained from Definition 4 by replacing the rule for existential quantification with:

- $\mathcal{M} \models_X \exists x\phi$ iff there exists $G : X \to \mathcal{P}(M) \setminus \{\emptyset\}$ such that $\mathcal{M} \models_{X[G/x]} \phi$.

Another way to modify the team semantics of Definition 4 is to require that the two subteams of the disjunction rule are disjoint. *Strict semantics* is now obtained from Definition 4 by replacing the disjunction rule with:

- $\mathcal{M} \models_X \phi \vee \psi$ iff there exist $Y, Z \subseteq X$, $Y \cup Z = X$, $Y \cap Z = \emptyset$, such that $\mathcal{M} \models_Y \phi$ and $\mathcal{M} \models_Z \psi$.

From now on we will consider lax semantics unless otherwise stated. Let $\mathcal{C}$ be a subset of $\{=(\ldots), \perp_c, \subseteq\}$. We denote by $\mathrm{FO}(\mathcal{C})$ (omitting the set parenthesis of $\mathcal{C}$) the logic obtained by adding the atoms of $\mathcal{C}$ to first-order logic. First note that for $\mathrm{FO}(=(\ldots), \perp_c, \subseteq)$ the locality principle holds.

Proposition 10 (Locality)**.** *Suppose* $\phi \in \mathrm{FO}(=(\ldots), \perp_c, \subseteq)$ *and* V *are such that* $Fr(\phi) \subseteq V$*. Then*

$$\mathcal{M} \models_X \phi \text{ iff } \mathcal{M} \models_{X \restriction V} \phi.$$

However, under strict semantics neither $\mathrm{FO}(\subseteq)$ nor $\mathrm{FO}(\perp_c)$ satisfy locality. To illustrate this in $\mathrm{FO}(\subseteq)$, let $\phi := x \subseteq y \vee z \subseteq y$ and let X be as in Figure 1. Then clearly $X \models \phi$.[1] However, $X \restriction \{x, y, z\} \not\models \phi$ since $X \restriction \{x, y, z\}$ collapses as

	x	y	z	v
s_0	0	1	2	3
s_1	1	0	1	3
s_2	1	0	1	4
s_3	2	1	0	4

Figure 1. X

illustrated in Figure 2.

	x	y	z
s_0	0	1	2
s_1	1	0	1
s_3	2	1	0

Figure 2. $X \restriction \{x, y, z\}$

Also neither $\mathrm{FO}(\subseteq)$ nor $\mathrm{FO}(\perp_c)$ is downwards closed. To illustrate this in $\mathrm{FO}(\perp_c)$, we first choose a simple pure independence atom $x \perp y$. A *pure independence atom* is an atom of the form $\vec{x} \perp_{\emptyset} \vec{y}$, written as a shorthand $\vec{x} \perp \vec{y}$. Since Y of Figure 3 satisfies $x \perp y$ but its subteam $Y \setminus \{s_3\}$ does not, $\mathrm{FO}(\perp_c)$ is not downwards closed.

For logics $\mathcal{L}$ and $\mathcal{L}'$ and a vocabulary τ, we write $\mathcal{L}[\tau] \leq \mathcal{L}'[\tau]$ if every $\mathcal{L}[\tau]$-sentence is logically equivalent to some $\mathcal{L}'[\tau]$-sentence. We also write $\mathcal{L} \leq \mathcal{L}'$ if

[1] We use $X \models \phi$ as a shorthand for $\mathcal{M} \models_X \phi$ if the truth of ϕ depends only on the team X.

	x	y
s_0	0	0
s_1	1	1
s_2	0	1
s_3	1	0

FIGURE 3. Y

$\mathcal{L}[\tau] \leq \mathcal{L}'[\tau]$ for all vocabularies τ. Equality and inequality relations are obtained from $\leq$ naturally. The following theorem characterizes the expressive powers of $\mathrm{FO}(\perp_c)$ and $\mathrm{FO}(\subseteq)$. For $\mathrm{FO}(\perp_c)$, the result is no surprise, but for $\mathrm{FO}(\subseteq)$ we obtain two different characterizations, depending on which version of team semantics is being used. Under lax semantics $\mathrm{FO}(\subseteq)$ captures positive greatest fixed-point logic (GFP^+) but, as we show in Paper I, under strict semantics its expressive power coincides with that of ESO.

Theorem 11 ([20, 23]).
- $\mathrm{FO}(\perp_c) = \mathrm{ESO}$,
- $\mathrm{FO}(\subseteq) = \mathrm{GFP}^+$,
- $\mathrm{FO}(\subseteq) = \mathrm{ESO}$ *(under strict semantics).*

Also on the level of open formulae, $\mathrm{FO}(\perp_c)$ captures all ESO definable classes of teams [18], being hence strictly more expressive than $\mathrm{FO}(=(\ldots))$ (note that by Proposition 7 even the simple pure independence atom $x \perp y$ is not definable in $\mathrm{FO}(=(\ldots))$). It is also worth noting that $\mathrm{FO}(\perp)$, that is FO extended with only pure independence atoms, already captures the full expressive power of $\mathrm{FO}(\perp_c)$, both on the level of sentences and open formulae [21]. From Theorem 11, using classical results in descriptive complexity theory, we obtain the following corollary.

Corollary 12 ([33, 46, 14]).
- *A class $\mathcal{C}$ of finite linearly ordered models is definable in* $\mathrm{FO}(\subseteq)$ *iff it can be recognized in* PTIME.
- *A class $\mathcal{C}$ of finite models is definable in* $\mathrm{FO}(\perp_c)$ *iff it can be recognized in* NP.

5. Expressivity Hierarchies

In this section we consider the first part of the thesis. The idea is to examine whether there exist some natural parameters of independence and inclusion logic that give rise to infinite expressivity or complexity hierarchies. As an example of such a parameter, note that many logics have some notion of arity. In second-order logic this arises from arity of quantified relations, and in fixed-point logic from arity of fixed-point operators. A natural question is then to ask whether in these situations increase in arity implies also increase in expressive power or complexity. In this section we survey Papers I-III that examine analogous questions in independence and inclusion logic. Two main parameters are considered. These are arity of a dependency atom and number of universal quantifiers. By restricting either of these parameters we define different fragments of independence and inclusion logic. Then relating these fragments to their analogues in existential second-order logic or fixed-point logics we show that they generate infinite expressivity and complexity hierarchies. When considering number of universal quantifiers, we notice that the corresponding expressivity hierarchies sometimes collapse at a low level. These results in turn follow from properties of independence and inclusion atoms.

The following fragments are studied:

Definition 13. Let $\mathcal{C}$ be a subset of $\{=(\ldots), \bot_c, \bot, \subseteq\}$ and let $k \in \mathbb{N}$. Then

(1) $\mathrm{FO}(\mathcal{C})(k{-}\mathrm{dep})$ is the class of sentences of $\mathrm{FO}(\mathcal{C})$ in which dependence atoms of the form $=(\vec{z}, y)$, where $\vec{z}$ is of length at most k, may appear.
(2) $\mathrm{FO}(\mathcal{C})(k{-}\mathrm{ind})$ is the class of sentences of $\mathrm{FO}(\mathcal{C})$ in which independence atoms of the form $\vec{y}\bot_{\vec{x}}\vec{z}$, where $\vec{x}\vec{y}\vec{z}$ has at most $k+1$ distinct variables, may appear.
(3) $\mathrm{FO}(\mathcal{C})(k{-}\mathrm{inc})$ is the class of sentences of $\mathrm{FO}(\mathcal{C})$ in which inclusion atoms of the form $\vec{a} \subseteq \vec{b}$, where $\vec{a}$ and $\vec{b}$ are of length at most k, may appear.
(4) $\mathrm{FO}(\mathcal{C})(k\forall)$ is the class of sentences of $\mathrm{FO}(\mathcal{C})$ in which every variable is quantified exactly once and at most k universal quantifiers occur.

For an increasing (with respect to $\leq$) sequence of logics $(\mathcal{L}_k)_{k\in\mathbb{N}}$, we say that the $\mathcal{L}_k$-hierarchy collapses at level m if $\mathcal{L}_m = \bigcup_{k\in\mathbb{N}} \mathcal{L}_k$. If the $\mathcal{L}_k$-hierarchy does not collapse at any level, then we say that the hierarchy is infinite. An infinite $\mathcal{L}_k$-hierarchy is called strict if $\mathcal{L}_k < \mathcal{L}_{k+1}$ for all $k \in \mathbb{N}$.

5.1. **Universal Hierarchies.** The background for studying fragments defined by restricting the number of universal quantifiers is in analogous ESO fragments.We let $\mathrm{ESO}_f(k\forall)$ be the class of ESO sentences in Skolem normal form

$$\exists f_1 \ldots \exists f_n \forall x_1 \ldots \forall x_l \psi$$

where $l \leq k$, and ψ is a quantifier-free formula. In [25] Grandean and Olive showed that $\mathrm{ESO}_f(k\forall)$ captures the complexity class $\mathsf{NTIME}_{\mathrm{RAM}}(n^k)$. This means that classes of models $\mathfrak{A}$ definable by an $\mathrm{ESO}_f(k\forall)$ sentence are exactly those that can be recognized by a non-deterministic random acces machine in time $O(n^k)$ where n is the size of $\mathrm{Dom}(\mathfrak{A})$. From Cook's hierarchy theorem for non-deterministic polynomial time, it follows that $\mathsf{NTIME}_{\mathrm{RAM}}(n^k)$, and hence also $\mathrm{ESO}_f(k\forall)$, form a strict hierarchy [10]. In [12] Kontinen and Durand related $\mathrm{FO}(=(\ldots))(k\forall)$ to $\mathrm{ESO}_f(k\forall)$ and showed that

Theorem 14 ([12])**.** $\mathrm{FO}(=(\ldots))(k\forall) < \mathrm{FO}(=(\ldots))((2k+2)\forall)$.

In Papers I-III we continue in this line by considering analogous hierarchies in independence and inclusion logic. In Paper II we study the fragments $\mathrm{FO}(\subseteq)(k\forall)$ and $\mathrm{FO}(\bot_c)(k\forall)$ under strict semantics. For $\mathrm{FO}(\bot_c)$ we obtain an infinite hierarchy, similar to that of $\mathrm{FO}(=(\ldots))$. We show that $\mathrm{FO}(\bot_c)(k\text{-ind}) \leq \mathrm{ESO}_f((k+1)\forall)$ and $\mathrm{ESO}_f(k\forall) \leq \mathrm{FO}(\bot_c)(2k\forall)$, thus obtaining the following theorem.

Theorem 15 (II)**.** *Under strict semantics,* $\mathrm{FO}(\perp_{\mathrm{c}})(k\text{-ind}) < \mathrm{FO}(\perp_{\mathrm{c}})((2k+4)\text{-ind})$*.*

Surprisingly though, we show that $\mathrm{FO}(\subseteq)(k\forall)$ captures $\mathrm{ESO}_f(k\forall)$ and hence also $\mathsf{NTIME}_{\mathrm{RAM}}(n^k)$. This gives rise to a strict expressivity hierarchy.

Theorem 16 (II)**.** *Under strict semantics,* $\mathrm{FO}(\subseteq)(k\forall) < \mathrm{FO}(\subseteq)((k+1)\forall)$*.*

Note that Theorems 14-16 hold for any vocabulary. In Papers I and III, we show that under lax semantics the universal hierarchies for $\mathrm{FO}(\subseteq)$ and $\mathrm{FO}(\perp)$ collapse at levels 1 and 2, respectively:

Theorem 17 (I,III)**.**

- $\mathrm{FO}(\subseteq)(1\forall) = \mathrm{FO}(\subseteq)$,
- $\mathrm{FO}(\perp)(2\forall) = \mathrm{FO}(\perp)$.

The idea behind these two results is that universal quantification $\forall x$ can be simulated by existential quantification $\exists x$ together with inclusion (or independence) atoms that force one to extend assignments of a team with all possible evaluations of x.

5.2. **Arity Hierarchies.** Let us then consider arity hierarchies. For results regarding independence logic the background is again in existential second-order logic. Ajtai showed that even cardinality of a $k+1$-ary relation cannot be expressed as an ESO sentence where quantified relations are at most k-ary [2]. In [12] and Paper I this result is related to arity fragments of dependence and independence logic, respectively. If $\mathrm{ESO}_f(k\text{-ary})$ is the class of ESO sentences in which at most k-ary relations and functions are quantified, then we have the following arity correspondence between ESO, $\mathrm{FO}(=(\ldots))$ and $\mathrm{FO}(\perp_{\mathrm{c}})$.

Theorem 18 ([12], I)**.** $\mathrm{ESO}_f(k\text{-ary}) = \mathrm{FO}(=(\ldots))(k\text{-dep}) = \mathrm{FO}(\perp_{\mathrm{c}})(k\text{-ind})$*.*

Then using Ajtai's result (that extends to $\mathrm{ESO}_f(k\text{-ary})$, see [13]) we obtain a strict arity hierarchy for our logics.

Theorem 19 ([2, 12], I)**.**

- $\mathrm{FO}(=(\ldots))(k\text{-dep}) < \mathrm{FO}(=(\ldots))(k+1\text{-dep})$,
- $\mathrm{FO}(\perp_{\mathrm{c}})(k\text{-ind}) < \mathrm{FO}(\perp_{\mathrm{c}})(k+1\text{-ind})$.

However, it remains open whether these hierarchies are strict over all vocabularies τ. By Theorem 18, solving these questions would have consequences beyond dependence or independence logic. In the special case where τ is empty strictness would imply a strict arity hierarchy within spectra of first-order sentences (a *spectrum* of a first-order sentence is the set of cardinalities of its finite models). This question of whether the *spectrum arity hierarchy* is strict or not was left open in [15] and is still unanswered. It is not even known whether there exists any spectrum that is not the spectrum of a first-order sentence over one binary relation symbol. These questions and their analogues regarding *generalized spectra* (i.e. ESO definable classes of finite models) have been studied by Fagin in [15] (see also [17]). So far for ESO, FO(=(...)) and FO($\perp_c$) we know from Ajtai's result that for all vocabularies τ there is a strict arity hierarchy up to k whenever τ contains a k-ary relation symbol.

In Paper III we study hierarchies in inclusion logic. We prove the following srict arity hierarchy.

Theorem 20 (III)**.** $\mathrm{FO}(\subseteq)(k\text{-inc}) < \mathrm{FO}(\subseteq)(k+1\text{-inc})$.

The proof of Theorem 20 is based on Grohe's work in [26] where he proved a strict arity hierarchy for transitive closure logic (TC), least fixed-point logic (LFP), inflationary fixed-point logic (IFP) and partial fixed-point logic (PFP). More precisely, it was shown that

$$\mathrm{TC}^k \not\leq \mathrm{PFP}^{k-1} \tag{21}$$

where the superscript part gives the maximum arity allowed for the fixed-point operator. Since $\mathrm{TC}^k \leq \mathrm{LFP}^k \leq \mathrm{IFP}^k \leq \mathrm{PFP}^k$, a strict arity hierarchy is obtained for each of these logics. However, unlike the previous case, this does not directly lead to a proof of Theorem 20 because translations between inclusion logic and fixed-point logics do not seem to respect arities (see [20]). Hence we show (20) in a more constructive way. First we define a first-order formula indicating that the k-tuples $\vec{x}$ and $\vec{y}$ form a $2k$-clique in a graph. That is, we define $\mathrm{EDGE}_k(\vec{x},\vec{y})$ as follows:

$$\mathrm{EDGE}_k(\vec{x},\vec{y}) := \bigwedge_{1\leq i,j\leq k} E(x_i,y_j) \wedge \bigwedge_{1\leq i\neq j\leq k} (E(x_i,x_j) \wedge E(y_i,y_j)).$$

Then we take negation of the transitive closure of $\mathrm{EDGE}_k(\vec{b},\vec{c})$ where $\vec{b}$ and $\vec{c}$ are k-ary constant sequences, and show that this property is definable in $\mathrm{FO}(\subseteq)(k\text{-inc})$ but not in $\mathrm{FO}(\subseteq)(k-1\text{-inc})$ (the same property was also used to show (21)). The definability in $\mathrm{FO}(\subseteq)(k\text{-inc})$ follows from Galliani's observation in [18], and the non-definability in $\mathrm{FO}(\subseteq)(k-1\text{-inc})$ is showed by applying results of [26] in the team semantics setting. In addition to this we show in Paper I that k-ary inclusion logic translates into k-ary independence logic.

Theorem 22 (I)**.** $\mathrm{FO}(\subseteq)(k\text{-inc}) < \mathrm{FO}(\perp_c)(k\text{-ind})$.

Note that the inequality in the previous theorem is strict since $\mathrm{FO}(\perp_c)(1\text{-ind})$ can already express properties that are inexpressible in $\mathrm{FO}(\subseteq)$. For instance, even cardinality of finite models is definable in $\mathrm{FO}(\perp_c)(1\text{-ind})$ since $\mathrm{FO}(\perp_c)(k\text{-ind}) = \mathrm{ESO}_f(k\text{-ary})$, but cannot be expressed in $\mathrm{FO}(\subseteq)$ by the result of [20].

An interesting open question is whether arity fragments of inclusion logic can be related to subclasses of PTIME if one restricts attention to finite ordered models. In this restricted case we know that arity fragments of PFP form a strict hierarchy since they can be related to the degree hierarchy within PSPACE. However, collapse of the arity hierarchy for IFP (or LFP) implies PTIME < PSPACE and strictness implies LOGSPACE < PTIME [32]. It remains open whether settling these questions for inclusion logic have such consequences.

6. Axiomatizations

The search for axiomatizations describing (as completely as possible) truth in a mathematical structure or validities of a formal language is fundamental in mathematics and computer science with dependence logic being no exception. In this section we consider Papers IV-VI that contain results regarding axiomatizability of different variants of dependence logic. Although the set of valid dependence or independence logic sentences is highly undecidable, meaning that no sound and complete axiomatization exists, certain possibilities remain still open. One possibility is to consider an analogue of Henkin semantics. This reduces complexity to the level of first-order logic, thus enabling complete recursively enumerable axiomatizations. Galliani has taken this direction for independence logic in [19]. Another strategy is to search for axiomatizations that capture logical consequence completely with respect to some fragment of a logic. For instance, it is a recursively enumerable problem to decide whether a first-order sentence follows from an arbitrary dependence logic sentence. This follows from the fact that dependence logic sentences translate into ESO. Therefore, it is possible to define axioms that are complete with respect to all first-order consequences of $\mathrm{FO}(=(\dots))$ sentences, and such an axiomatization has been presented in [36]. In Paper IV we generalize this approach by presenting an axiomatization that is complete with respect to all first-order consequences of $\mathrm{FO}(\perp_{\mathrm{c}})$ sentences. Namely, for a set of $\mathrm{FO}(\perp_{\mathrm{c}})$ sentences $T \cup \{\phi\}$, we write $T \models \phi$ if ϕ is true in all models that satisfy the sentences of T, and $T \vdash_{\mathcal{I}} \phi$ if ϕ is provable from T in our deduction system. Then we prove the following.

Theorem 23 (IV)**.** *Let $T \cup \{\phi\}$ be a set of sentences of* $\mathrm{FO}(\perp_{\mathrm{c}})$ *where ϕ is first-order. Then $T \models \phi$ iff $T \vdash_{\mathcal{I}} \phi$.*

Another scenario occurs if we restrict attention to atomic fragments of our logics. This is particularly interesting since studying these fragments relates directly to the theory of dependencies in relational data models. Before introducing the results of Papers V-VI, let us survey this connection shortly. The research of database dependencies initiated in the early 1970s by the introduction of relational models and functional dependencies [8, 9]. Soon many different classes of dependencies were presented and the field saw a rapid explosion (for instance, a survey from 1986 gives references to more than 600 research papers on dependency theory [44]). Some of the new dependencies were motivated by practical examples while some had a pure theoretical motivation. The most important dependencies introduced include functional, inclusion, join and multivalued dependencies. Many of these have been also studied extensively in the team semantics setting as variants of dependence logic [18, 23, 45].

In the 1980s it was observed that most of the considered dependencies are expressible in first-order logic [40]. These expressions can be usually written in the form:

$$\forall x_1 \dots \forall x_n \phi(x_1, \dots, x_n) \to \exists z_1 \dots \exists z_k \psi(y_1, \dots, y_m), \tag{24}$$

where $\{z_1, ..., z_k\} = \{y_1, ..., y_m\} \setminus \{x_1, ..., x_n\}$, and ϕ is a possibly empty conjuntion and ψ is a non-empty conjunction of atoms. In both ϕ and ψ one finds relational atoms of the form $R(w_1, \dots, w_l)$ and equality atoms of the form $w = w'$ where each of $w, w', w_1, \dots, w_l$ is a variable. Dependencies of the form (24) are in general called *embedded dependencies*, and in absence of existentially quantified variables z_i they are called *full dependencies*.

This division to embedded and full dependencies is important when trying to understand the implication problem for various classes of dependencies. An *implication problem* is given as a set $\Sigma \cup \{\sigma\}$ of dependencies, and the problem is to decide whether σ is true in all relations that satisfy Σ (written $\Sigma \models \sigma$). A finite implication problem instead, is to decide whether the above is true over all finite relations (written $\Sigma \models_{\text{FIN}} \phi$). If $\Sigma = \{\sigma_1, \dots, \sigma_n\}$, then $\Sigma \models \sigma$ (or $\Sigma \models_{\text{FIN}} \sigma$) iff there is no unrestricted (finite) model of $\sigma_1 \wedge \ldots \wedge \sigma_n \wedge \neg\sigma$. Restricting attention to full dependencies only, we notice that the above sentence can be rewritten in prenex normal form where the quantifier prefix has the form $\exists^*\forall^*$. The class of sentences of this form (and with no function symbols) is called the Bernays-Schönfinkel class [4], and it is straightforward to show that the unrestricted and finite satisfiability problems coincide for this class. Hence it follows that the two problems are decidable (actually NEXPTIME-complete [37]), and therefore we obtain that the implication problem for full dependencies is decidable.

For embedded dependencies the situation is different since already for the subclass of embedded multivalued dependencies the implication problem is undecidable [28, 29]. An EMVD $X \twoheadrightarrow Y|Z$ is given for attribute sets X, Y, Z and is satisfied by a relation r if for all tuples $t, t' \in r$ that agree on X there exists $t'' \in r$ that agrees with t on XY and with t' on $Z \setminus XY$. Since their implication is undecidable it also follows that they lack finite axiomatization, meaning that no finite sound and complete set of Horn rules exists for EMVDs (otherwise a decision procedure would follow). However, this does not mean that EMVDs cannot be axiomatized within a larger class of dependencies (a trivial example uses inference rules of first-order logic). Hence one strategy in such situations has been to search for inference rules that are complete for some generalized class of dependencies. This has in part lead to a multitude of different dependency notions [1].

In Paper V we present an axiomatization for embedded multivalued dependencies together with inclusion dependencies but we use a slightly different approach and follow Mitchell's axiomatization for the implication problem of functional and inclusion dependencies taken together [38]. It is worth noting that this problem is undecidable even though the implication problem for FDs (or INDs) alone is decidable and enjoys finite axiomatization [3, 6, 7, 39]. The key axiom in [38] is the following:

- Attribute Introduction:

$$\text{if } U \subseteq V \wedge V \to B, \text{ then } UA \subseteq VB.$$

In this rule A is called *new* which means that it is not allowed to appear in any assumption or in any previous step of a deduction. The idea is that this new attribute is to be thought of as implicitly existentially quantified. Although the axiomatization in [38] is finite, the corresponding decision procedure does not necessarily terminate since there is no *a priori* limit for the number of new attributes needed in a deduction. In Paper V we take a similar approach to this and present the following axioms for conditional independence (i.e. EMVDs) and inclusion dependencies taken together.

Definition 25.

(1) Reflexivity:
$$\vec{x} \subseteq \vec{x}.$$
(2) Projection and Permutation:
$$\text{if } x_1 \dots x_n \subseteq y_1 \dots y_n, \text{ then } x_{i_1} \dots x_{i_k} \subseteq y_{i_1} \dots y_{i_k},$$
for each sequence $i_1, \dots, i_k$ of integers from $\{1, \dots, n\}$.

(3) Transitivity:
$$\text{if } \vec{x} \subseteq \vec{y} \wedge \vec{y} \subseteq \vec{z}, \text{ then } \vec{x} \subseteq \vec{y}.$$
(4) Identity Rule:
$$\text{if } ab \subseteq cc \wedge \phi, \text{ then } \phi',$$
where ϕ' is obtained from ϕ by replacing any number of occurrences of a by b.

(5) Inclusion Introduction:
$$\text{if } \vec{a} \subseteq \vec{b}, \text{ then } \vec{a}x \subseteq \vec{b}c,$$
where x is a *new* variable.

(6) Start Axiom:
$$\vec{a}\vec{c} \subseteq \vec{a}\vec{x} \wedge \vec{b} \perp_{\vec{a}} \vec{x} \wedge \vec{a}\vec{x} \subseteq \vec{a}\vec{c}$$
where $\vec{x}$ is a sequence of pairwise distinct *new* variables.

(7) Chase Rule:
$$\text{if } \vec{y} \perp_{\vec{x}} \vec{z} \wedge \vec{a}\vec{b} \subseteq \vec{x}\vec{y} \wedge \vec{a}\vec{c} \subseteq \vec{x}\vec{z}, \text{ then } \vec{a}\vec{b}\vec{c} \subseteq \vec{x}\vec{y}\vec{z}.$$
(8) Final Rule:
$$\text{if } \vec{a}\vec{c} \subseteq \vec{a}\vec{x} \wedge \vec{b} \perp_{\vec{a}} \vec{x} \wedge \vec{a}\vec{b}\vec{x} \subseteq \vec{a}\vec{b}\vec{c}, \text{ then } \vec{b} \perp_{\vec{a}} \vec{c}.$$

Rules 5 and 6 are analogous to Attribute Introduction in the sense that they introduce new variables that are not allowed to appear in assumptions or in earlier proof steps. However, the novelty here is that these new variables should be interpreted as existentially quantified in the lax semantics sense. With the so-called strict existential quantification (see item 4 of Definition 4), Rule 5 is not sound. We write $\Sigma \vdash \phi$ if ϕ does not contain any new variables and can be deduced from Σ using the rules of Definition 25. We then show the following theorem.

Theorem 26 (V)**.** *Let $\Sigma \cup \{\phi\}$ be a finite set of conditional independence and inclusion atoms. Then $\Sigma \models \phi$ iff $\Sigma \vdash \phi$.*

In Paper VI we consider another implication problem that combines two sorts of different dependencies. Namely, we examine the interaction between keys and (pure) independence atoms. Both of these atoms are important in everyday practise of data processing, and have hence received detailed interest from the database research community since the 1970s [5, 11, 16, 22, 34, 41]. Although their interarction has not yet been studied, it is known that both classes in isolation enjoy simple axiomatizations. Our contribution is to show that the situation changes drastically when these classes are considered together. At first, we show that their finite and unrestricted implication problems differ from each other. Secondly, we show that the finite implication problem has no finite axiomatization. The question of whether the finite and unrestricted implication problems are decidable remains open. However, understanding axiomatizability can be seen as a first step towards a solution. For instance it was shown in [42] that embedded multivalued dependencies lack finite axiomatization both for the finite and unrestricted implication, and later it was shown that both these problems are undecidable [28, 29]. On the other hand, lack of a finite axiomatization does not always mean that the corresponding implication problem is undecidable. Join dependencies are instances of full dependencies and hence their implication is decidable. However, it is known that they cannot be finitely axiomatized [43].

Keys and (pure) independence atoms are now defined as follows. Let R be a relation schema i.e. a set of attributes, and let $X \subseteq R$. Then $\mathcal{K}(X)$ is an *R-key*, given the following semantic rule for a relation r over R:

- $r \models \mathcal{K}(X)$ if and only if for all $t, t' \in r$: if $t(X) = t'(X)$, then $t = t'$.

Also, if $X, Y \subseteq R$, then $X \perp Y$ is an *R-independence atom*, given the following semantic rule:

- $r \models X \perp Y$ if and only if for all $t, t' \in r$ there exists a $t'' \in r$ such that $t''(X) = t(X) \wedge t''(Y) = t'(Y)$.

A simple example shows that if keys and independence atoms are taken together, then the unrestricted and finite implication problem differ from one another. This holds already in the case where independence atoms are allowed to be at most unary i.e. of the form $X \perp Y$ where X and Y are single attributes.

Theorem 27 (VI)**.** *Let $R := \{A, B, C, D\}$ and $\Sigma := \{A \perp B, C \perp D, \mathcal{K}(AD), \mathcal{K}(BC)\}$. Then $\Sigma \models_{\mathrm{FIN}} \mathcal{K}(AB)$ and $\Sigma \not\models \mathcal{K}(AB)$.*

Already in this restricted case, the finite implication problem cannot have a k-ary axiomatization. A k-ary axiomatization consists of at most k-ary Horn rules i.e. rules of the form $\sigma_1 \wedge \ldots \wedge \sigma_n \rightarrow \tau$ where $n \leq k$.

Theorem 28 (VI)**.** *For no natural number k, there exists a sound and complete k-ary axiomatization of the finite implication problem for keys and unary independence atoms taken together.*

The general implication problem is however finitely axiomatizable. We write $\Sigma \vdash_{\mathfrak{J}} \phi$ if ϕ can be deduced from Σ using the rules of Table 1.

Theorem 29 (VI)**.** *Assume that R is a relation schema and $\Sigma \cup \{\phi\}$ consists of R-keys and unary R-independence atoms. Then $\Sigma \vdash_{\mathfrak{J}} \phi$ iff $\Sigma \models \phi$.*

$\dfrac{}{\emptyset \perp X}$ (trivial independence, $\mathcal{R}1$)	$\dfrac{X \perp Y}{Y \perp X}$ (symmetry, $\mathcal{R}2$)	$\dfrac{X \perp X \quad Y \perp Z}{XY \perp Z}$ (constancy, $\mathcal{R}3$)
$\dfrac{X \perp YZ}{X \perp Y}$ (decomposition, $\mathcal{R}4$)	$\dfrac{X \perp Y \quad XY \perp Z}{X \perp YZ}$ (exchange, $\mathcal{R}5$)	$\dfrac{}{\mathcal{K}(R)}$ (trivial key, $\mathcal{R}6$)
$\dfrac{\mathcal{K}(X)}{\mathcal{K}(XY)}$ (upward closure, $\mathcal{R}7$)	$\dfrac{X \perp X \quad \mathcal{K}(XY)}{\mathcal{K}(Y)}$ (1st composition, $\mathcal{R}8$)	$\dfrac{X \perp Y \quad \mathcal{K}(X)}{Y \perp Y}$ (2nd composition, $\mathcal{R}9$)

Table 1. Axiomatization $\mathfrak{J}$ of Independence Atoms and Keys in Database Relations

Note that Theorem 28 generalizes to the case where independence atoms have no arity restrictions. This concludes the preview of the results of the thesis.

References

[1] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

[2] Miklos Ajtai. Σ^1_1-formulae on finite structures. *Ann. Pure Appl. Logic*, 4(1):1 – 48, 1983.

[3] William W. Armstrong. Dependency Structures of Data Base Relationships. In *Proc. of IFIP World Computer Congress*, pages 580–583, 1974.

[4] Paul Bernays and Moses Schönfinkel. Zum entscheidungsproblem der mathematischen logik. *Mathematische Annalen*, 99(1):342–372, 1928.

[5] Andrea Calì, Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. Data integration under integrity constraints. *Inf. Syst.*, 29(2):147–163, 2004.

[6] Marco A. Casanova, Ronald Fagin, and Christos H. Papadimitriou. Inclusion dependencies and their interaction with functional dependencies. In *Proceedings of the 1st ACM SIGACT-SIGMOD symposium on Principles of database systems*, PODS '82, pages 171–176, New York, NY, USA, 1982. ACM.

[7] Ashok K. Chandra and Moshe Y. Vardi. The implication problem for functional and inclusion dependencies is undecidable. *SIAM Journal on Computing*, 14(3):671–677, 1985.

[8] Edgar F. Codd. A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387, June 1970.

[9] Edgar F. Codd. Relational completeness of database sublanguages. In R. Rustin, editor, *Data Base Systems*, pages 65–98. Prentice-Hall, 1972.

[10] Stephen A. Cook. A hierarchy for nondeterministic time complexity. In *Conference Record, Fourth Annual ACM Symposium on Theory of Computing*, pages 187–192. ACM, 1972.

[11] János Demetrovics, Gyula O. H. Katona, Dezsö Miklós, Oleg Seleznjev, and Bernhard Thalheim. Asymptotic properties of keys and functional dependencies in random databases. *Theor. Comput. Sci.*, 190(2):151–166, 1998.

[12] Arnaud Durand and Juha Kontinen. Hierarchies in dependence logic. *ACM Transactions on Computational Logic (TOCL)*, 13(4):31, 2012.

[13] Arnaud Durand, Clemens Lautemann, and Thomas Schwentick. Subclasses of binary NP. *J. Log. Comput.*, 8(2):189–207, 1998.

[14] Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In *Complexity of Computation, SIAM-AMS Proceedings,*, volume Vol. 7, pages 43–73, 1974.

[15] Ronald Fagin. A spectrum hierarchy. *Mathematical Logic Quarterly*, 21(1):123–134, 1975.

[16] Ronald Fagin. A normal form for relational databases that is based on domains and keys. *ACM Trans. Database Syst.*, 6(3):387–415, 1981.

[17] Ronald Fagin. Finite-model theory - a personal perspective. *Theoretical Computer Science*, 116:3–31, 1993.

[18] Pietro Galliani. Inclusion and exclusion dependencies in team semantics: On some logics of imperfect information. *Annals of Pure and Applied Logic*, 163(1):68 – 84, 2012.

[19] Pietro Galliani. General models and entailment semantics for independence logic. *Notre Dame Journal of Formal Logic*, 54(2):253–275, 2013.

[20] Pietro Galliani and Lauri Hella. Inclusion Logic and Fixed Point Logic. In Simona Ronchi Della Rocca, editor, *Computer Science Logic 2013 (CSL 2013)*, volume 23 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 281–295, Dagstuhl, Germany, 2013. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

[21] Pietro Galliani and Jouko A. Väänänen. On dependence logic. In A. Baltag and S. Smets, editors, *Johan van Benthem on Logical and Informational Dynamics*, pages 101–119. Springer, 2014.

[22] Dan Geiger, Azaria Paz, and Judea Pearl. Axioms and algorithms for inferences involving probabilistic independence. *Information and Computation*, 91(1):128–141, 1991.

[23] Erich Grädel. Model-checking games for logics of imperfect information. *Theor. Comput. Sci.*, 493:2–14, 2013.

[24] Erich Grädel and Jouko Väänänen. Dependence and independence. *Studia Logica*, 101(2):399–410, 2013.

[25] Etienne Grandjean and Frédéric Olive. Graph properties checkable in linear time in the number of vertices. *J. Comput. Syst. Sci.*, 68(3):546–597, 2004.

[26] Martin Grohe. Arity hierarchies. *Ann. Pure Appl. Logic*, 82(2):103–163, 1996.

[27] Leon Henkin. Some Remarks on Infinitely Long Formulas. In *Infinitistic Methods. Proc. Symposium on Foundations of Mathematics*, pages 167–183. Pergamon Press, 1961.

[28] Christian Herrmann. On the undecidability of implications between embedded multivalued database dependencies. *Information and Computation*, 122(2):221 – 235, 1995.

[29] Christian Herrmann. Corrigendum to "on the undecidability of implications between embedded multivalued database dependencies" [inform. and comput. 122(1995) 221-235]. *Inf. Comput.*, 204(12):1847–1851, 2006.

[30] Jaakko Hintikka and Gabriel Sandu. Informational independence as a semantic phenomenon. In J.E Fenstad, I.T Frolov, and R. Hilpinen, editors, *Logic, methodology and philosophy of science*, pages 571–589. Elsevier, 1989.

[31] Wilfrid Hodges. Compositional Semantics for a Language of Imperfect Information. *Journal of the Interest Group in Pure and Applied Logics*, 5 (4):539–563, 1997.

[32] Henrik Imhof. Computational aspects of arity hierarchies. In *Computer Science Logic, 10th International Workshop, CSL '96, Annual Conference of the EACSL, Utrecht, The Netherlands, September 21-27, 1996, Selected Papers*, pages 211–225, 1996.

[33] Neil Immerman. Relational queries computable in polynomial time. *Information and control*, 68(1):86–104, 1986.

[34] Juha Kontinen, Sebastian Link, and Jouko A. Väänänen. Independence in database relations. In Leonid Libkin, Ulrich Kohlenbach, and Ruy J. G. B. de Queiroz, editors, *WoLLIC*, volume 8071 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 2013.

[35] Juha Kontinen and Jouko Väänänen. On definability in dependence logic. *Journal of Logic, Language and Information*, 3(18):317–332, 2009.

[36] Juha Kontinen and Jouko A. Väänänen. Axiomatizing first-order consequences in dependence logic. *Ann. Pure Appl. Logic*, 164(11):1101–1117, 2013.

[37] Harry R. Lewis. Complexity results for classes of quantificational formulas. *Journal of Computer and System Sciences*, 21(3):317 – 353, 1980.

[38] John C. Mitchell. The implication problem for functional and inclusion dependencies. *Information and Control*, 56(3):154–173, 1983.

[39] John C. Mitchell. The implication problem for functional and inclusion dependencies. *Information and Control*, 56(3):154 – 173, 1983.

[40] J. M. Nicolas. First order logic formalization for functional, multivalued and mutual dependencies. In *Proceedings of the 1978 ACM SIGMOD International Conference on Management of Data*, SIGMOD '78, pages 40–46, New York, NY, USA, 1978. ACM.

[41] Jan Paredaens. The interaction of integrity constraints in an information system. *J. Comput. Syst. Sci.*, 20(3):310–329, 1980.

[42] D. Stott Parker, Jr. and Kamran Parsaye-Ghomi. Inferences involving embedded multivalued dependencies and transitive dependencies. In *Proceedings of the 1980 ACM SIGMOD International Conference on Management of Data*, SIGMOD '80, pages 52–57, New York, NY, USA, 1980. ACM.

[43] Sergey V. Petrov. Finite axiomatization of languages for representation of system properties: Axiomatization of dependencies. *Information Sciences*, 47(3):339 – 372, 1989.

[44] Bernhard Thalheim. A review of research on dependency theory in relational databases. In *Proceedings of the 9th International Seminar on Database Management Systems*, pages 136–169, 1986.

[45] Jouko Väänänen. *Dependence Logic*. Cambridge University Press, 2007.

[46] Moshe Y. Vardi. The complexity of relational query languages. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 137–146. ACM, 1982.