TPLP **22** (5): 678–692, 2022. © The Author(s), 2022. Published by Cambridge University Press. 678
doi:10.1017/S1471068422000199 First published online 26 July 2022

*MV-Datalog+-: Effective Rule-based Reasoning with Uncertain Observations**

MATTHIAS LANZINGER
University of Oxford, Oxford OX1 2JD, UK
(*e-mail:* matthias.lanzinger@cs.ox.ac.uk)

STEFANO SFERRAZZA
University of Oxford, Oxford OX1 2JD, UK
Technische Universität Wien, Vienna, Austria
(*e-mail:* stefano.sferrazza@cs.ox.ac.uk)

GEORG GOTTLOB
University of Oxford, Oxford OX1 2JD, UK
(*e-mail:* georg.gottlob@cs.ox.ac.uk)

submitted 15 May 2022; accepted 8 June 2022

Abstract

Modern applications combine information from a great variety of sources. Oftentimes, some of these sources, like machine-learning systems, are not strictly binary but associated with some degree of (lack of) confidence in the observation. We propose MV-Datalog and MV-Datalog$^{\pm}$ as extensions of Datalog and Datalog$^{\pm}$, respectively, to the fuzzy semantics of infinite-valued Łukasiewicz logic Ł as languages for effectively reasoning in scenarios where such uncertain observations occur. We show that the semantics of MV-Datalog exhibits similar model theoretic properties as Datalog. In particular, we show that (fuzzy) entailment can be decided via minimal fuzzy models. We show that when they exist, such minimal fuzzy models are unique and can be characterised in terms of a linear optimisation problem over the output of a fixed-point procedure. On the basis of this characterisation, we propose similar many-valued semantics for rules with existential quantification in the head, extending Datalog$^{\pm}$.

KEYWORDS: Datalog, fuzzy logic programming, logic programming, Łukasiewicz, logic, uncertainty in AI, Datalog$^{\pm}$

1 Introduction

Datalog and its extensions have long been important languages in databases and at the foundation of many rule-based reasoning formalisms. Continuous theoretical and technical improvements have led to the development of highly efficient Datalog systems for widespread practical use in a variety of applications (see e.g, Maier *et al.* 2018).

However, in many such real-world applications the observations and facts that serve as the input are not actually certain but rather associated with some (possibly unknown)

* Stefano Sferrazza was supported by the Austrian Science Fund (FWF):P30930. Georg Gottlob is a Royal Society Research Professor and acknowledges support by the Royal Society in this role through the "RAISON DATA" project (Reference No. RP\R1\201074). Matthias Lanzinger acknowledges support by the Royal Society "RAISON DATA" project (Reference No. RP\R1\201074).

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

level of uncertainty. One particularly important situation where such a setting occurs are settings where observations are made by machine learning (ML) systems.

For example, say a database contains a relation that labels images of animals with the class of animal observed in the image. In modern settings, such labels are commonly derived from the output of ML systems that attempt to classify the image contents. Generally, such systems output possible labels for an image, together with a score that can be interpreted as the level of confidence in the labels correctness (cf., Mishkin *et al.* 2017; Lee *et al.* 2017).

When the labelling in our example is used in further logical inference, the level of confidence in the individual labels would ideally also be reflected in any knowledge inferred from these uncertain observations. However, classical logical reasoning provides no way to consider such information and requires the projection of uncertainty levels to be either true or false, usually via some simple numerical threshold. Beyond the overall loss of information, this can also lead to problematic outcomes where logical conclusions are true despite their derivation being based on observations made with only moderate confidence. With observations from ML systems becoming a commonplace feature of many modern reasoning settings, the projection to binary truth values is severely limiting the immense potential of integrating ML observations with logic programming.

Related Work There are two natural ways to interpret uncertain observations as described in the example above. One can consider them as the likelihood of the fact being true, that is, the level of confidence in a fact is interpreted as a probability of the fact holding true. Such an interpretation has been widely studied in the context of Problog (Raedt *et al.* 2007), Markov Logic Networks (Richardson and Domingos 2006), and probabilistic databases (Suciu *et al.* 2011; Cavallo and Pittarelli 1987). However, generally such formalisms make strong assumptions on the pairwise probabilistic independence of all tuples which can be difficult to satisfy. An approach to probabilistic reasoning in Datalog$^{\pm}$ based on the chase procedure was proposed very recently by Bellomarini *et al.* (2020).

Alternatively, one can express levels of confidence in a fact in terms of *degrees of truth* as in *fuzzy logics* (cf. Hájek 1998). That is, a fact that is considered to be true to a certain degree, in accordance with the level of confidence in the observation.

There have been a wide variety of approaches of combining logic programming with fuzzy logic. Pioneering work in this area by Achs and Kiss (1995), Ebrahim (2001), and Vojtás (2001) introduced a number of early foundational ideas for fuzzy logic programming. The first also considers fuzziness in the object domain (expressing the possibility that two constants may actually refer to the same object), via fuzzy similarity relations on the object domain. This idea was further expanded in the context of deductive databases by Iranzo and Sáenz-Pérez (2018).

One limiting factor of these approaches in our intended setting is the reliance on the *Gödel t-norm* (see, Preining 2010) as the basis for many-valued semantics. The Gödel t-norm of two formulas ϕ, ψ is the minimum of the respective truth degrees of ϕ and ψ. While this simplifies the model theory of the resulting logics, the resulting semantics are not ideal for our envisioned applications in which we want to combine uncertain observations from AI systems with large certain knowledge bases. Further discussion of the important differences between semantics based on the Gödel t-norm and the Łukasiewicz t-norm (see, Hájek 1998), which we propose to use in this paper, is provided in Section 2.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Finally, Probabilistic Soft Logic (PSL) was introduced by Bach *et al.* (2017) as a framework for fuzzy reasoning with semantics based on Łukasiewicz logic. PSL is a significantly different language with different use cases than our proposed languages. It allows for negation in rule bodies as well as disjunction in the head, in addition to certain forms of aggregation. Semantically it treats all rules as soft constraints and aims to find interpretations that satisfy rules as much as possible. Similar ideas have also been proposed in fuzzy description logics, see for example, Bobillo and Straccia (2016), Borgwardt and Peñaloza (2017), Stoilos *et al.* (2007), or Lukasiewicz and Straccia (2008) for an overview of fuzzy DLs.

Contributions We propose extensions of Datalog and Datalog$^{\pm}$, that is, Datalog with existential quantification in the head, to the many-valued semantics of infinite-valued Łukasiewicz logic. We argue that these languages offer natural semantics for practical applications with uncertain data while preserving key theoretical properties of the classical (two-valued) case. Overall, the contributions of this paper are summarised as follows.

- We introduce MV-Datalog as Datalog with Łukasiewicz semantics. We discuss adaptions of the standard minimal model concept to fuzzy models where every fact takes the least possible truth degree. We show that for consistent inputs, unique minimal models always exist and give a characterisation of such minimal models. Our results hold also for K-fuzzy models, that is, interpretations that satisfy formulas of the program only to degree K rather than fully.
- We show that MV-Datalog is suitable for practical applications. For one, our characterisation reveals that finding K-fuzzy models is feasible by building on standard methods as from the classical setting. In particular, we also show fuzzy fact entailment in MV-Datalog has the same complexity as fact entailment in Datalog.
- We propose MV-Datalog$^{\pm}$ as practice oriented extension of MV-Datalog by existential quantification in the head. To obtain intuitive semantics for logic programming we deviate from standard semantics of many valued existential quantification and instead argue for the use of strong existential quantification. We discuss fundamental model theoretic properties and challenges of MV-Datalog$^{\pm}$ and show how our characterisation result for MV-Datalog can be extended to capture a natural class of preferred models of MV-Datalog$^{\pm}$, providing the foundation for a system implementation of MV-Datalog$^{\pm}$.

Structure The rest of this paper is structured as follows. Terminology, notation and further preliminaries are stated in Section 2. Throughout Section 3, we introduce MV-Datalog and our main results for minimal fuzzy models in MV-Datalog. The further extension to MV-Datalog$^{\pm}$ is presented in Section 4. We conclude with final remarks and a brief discussion of future work in Section 5. Proof details that had to be omitted from the main text due to space restrictions are provided in the Appendix A of the arXiv version of this paper (cf., Lanzinger *et al.* 2022).

2 Preliminaries

We assume familiarity with standard notions of Datalog and first-order logic. Throughout this document, we consider only logical languages with no function symbols.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Datalog Fix a relational signature σ and a (countable) domain *Dom*. A *(relational) atom* is a term $R(u_1, \ldots, u_{\#R})$ where $\#R$ is the arity of relation symbol $R \in \sigma$ and for $0 \leq i \leq \#R$, u_i is a variable or a constant. If all positions of the atom are constants we say that it is a *ground atom* (or *fact*). A *(Datalog) rule* r is a first-order implication $\varphi \rightarrow \psi$ where φ is a conjunction of relational atoms and ψ consists of a single relational atom. We say that φ is the *body* of r (body(r)) and ψ is the *head* of r (head(r)). A rule is *safe* if all free variables in the head of a rule also occur in its body. A *Datalog program* is a set of safe Datalog rules. The models of a Datalog program are the first-order interpretations that satisfy all rules. For two models $\mathcal{I}, \mathcal{I}'$ of a Datalog program, we say that $\mathcal{I} \leq \mathcal{I}'$ if every every ground atom satisfied by $\mathcal{I}$ is also satisfied by $\mathcal{I}'$. The *minimal model* of a Datalog program Π is the model $\mathcal{I}$ of Π such that for all other models $\mathcal{I}'$ of Π it holds that $\mathcal{I} \leq \mathcal{I}'$.

A *database* D is a set of ground atoms, logically interpreted as a theory consisting of an atomic sentence for each ground atom. We say that $\mathcal{I}$ is a model of a program and a database (Π, D), if $\mathcal{I}$ is a model of both Π and D. We write $\Pi, D \models \varphi$ if formula φ is satisfied for every model of (Π, D). Note that this is equivalent to satisfaction under the minimal model of (Π, D).

Datalog$^{\pm}$ The extension of Datalog with existential quantification in the head, led to the well studied Datalog$^{\pm}$ (Calì *et al.* 2012). A *Datalog*$^{\pm}$ *rule* is either a Datalog rule or a first-order formula of the form $\varphi \rightarrow \exists \mathbf{y}\ \psi(\mathbf{y})$ where φ is again a conjunction of relational atoms and ψ is a relational atom that mentions all variables in $\mathbf{y}$. A *Datalog*$^{\pm}$ *program* is a set of safe Datalog$^{\pm}$ rules. Models in Datalog$^{\pm}$ are defined in the same way as for plain Datalog. Note that existential quantification ranges over the whole (infinite) domain *Dom* rather than the active domain of the program/database. It follows that, in contrast to Datalog, Datalog$^{\pm}$ programs do not have a unique minimal model. Furthermore, fact entailment in Datalog$^{\pm}$ is undecidable in general but vast decidable fragments are known.

Oblivious Chase Fact entailment in Datalog$^{\pm}$ (and Datalog) is closely linked to the so-called *chase* fixed-point procedures. In the following, we will be particularly interested in a specific chase procedure called the *oblivious chase*. We give only the minimal necessary technical details here and refer to Calì *et al.* (2013) for full details.

For any formula φ, we write vars(φ) for the set of *free variables* in the formula. Given a database D, we say that a Datalog$^{\pm}$ rule r is *obliviously applicable* to D if there exists a homomorphism $h\colon \text{vars}(r) \rightarrow \mathit{Dom}$ such that all atoms in $h(\text{body}(r))$ are in D.

Let r be a Datalog$^{\pm}$ rule that is obliviously applicable to D via a homomorphism h. Let h' be the extension of h to the existentially quantified variables $\mathbf{y}$ in head(r) such that for each $y \in \mathbf{y}$, $h'(y)$ is a fresh constant, called a *labelled null*, not occurring in D ($h' = h$ when r is a Datalog rule). Let ψ be the atomic formula in head(r). The result of the *oblivious application* of r, h to D is $D' = D \cup h'(\psi)$. We write $D \xrightarrow{r,h} D'$. Let D be a database and Π a Datalog$^{\pm}$ program. The *oblivious chase sequence* $\mathit{OChase}(\Pi, D)$ is the sequence $D_0, D_1, D_2, \ldots$ with $D = D_0$ such that for all $i \geq 0$, $D_i \xrightarrow{r_i,h_i} D_{i+1}$ and $r_i \in \Pi$, the chase sequence proceeds as long as there are oblivious applications r, h that have not yet been applied. We refer to the limit of the oblivious chase sequence as $\mathit{OLim}(\Pi, D)$. Every oblivious application r, h induces a grounding of r where all variables

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

(including the existentially quantified ones) are assigned constants according to h and any existential quantification is removed. We refer to the set of all ground rules induced in this way by the applications in $OChase(\Pi, D)$ as $OGround(\Pi, D)$. Note that for every Π and D, $OLim(\Pi, D)$ is unique up to isomorphism.

Łukasiewicz Logic While there are many semantics for many-valued logic, in this paper we will consider infinite-valued Łukasiewicz logic Ł (see, Hájek 1998). Some of our techniques are particular to Łukasiewicz logic and the results given in this paper generally do not apply to other many-valued logics.

For some relational signature σ, we consider the following logical language, where R is a relational atom (in σ) and a *formula* φ is defined via the grammar

$$\varphi ::= R \mid \varphi \odot \varphi \mid \varphi \oplus \varphi \mid \varphi \to \varphi \mid \neg\varphi.$$

For a signature σ and domain Dom, let $GAtoms$ be the the set of all ground atoms with respect to σ and Dom. A *truth assignment* is a function $\nu\colon GAtoms \to [0,1]$, intuitively assigning a degree of truth in the real interval $[0,1]$ to every ground atom. By slight abuse of notation, we also apply ν to formulas to express the truth of ground formulas γ, γ' according to the following inductive definitions.

$$\begin{aligned}
\nu(\neg\gamma) &= 1 - \nu(\gamma) \\
\nu(\gamma \odot \gamma') &= \max\{0, \nu(\gamma) + \nu(\gamma') - 1\} \\
\nu(\gamma \oplus \gamma') &= \min\{1, \nu(\gamma) + \nu(\gamma')\} \\
\nu(\gamma \to \gamma') &= \min\{1, 1 - \nu(\gamma) + \nu(\gamma')\}.
\end{aligned}$$

That is, $\odot$ is the usual Łukasiewicz *t-norm* and $\oplus$ is the corresponding *t-conorm*, which take the place of conjunction and disjunction, respectively. Implication $\gamma \to \gamma'$ could equivalently be defined as usual as $\neg\gamma \oplus \gamma'$. Note that De Morgan's rule holds as usual for $\odot$ and $\oplus$.

For rational $K \in (0,1]$ we say that a formula φ is *K-satisfied* by ν if for every grounding γ of φ over Dom it holds that $\nu(\gamma) \geq K$. Whenever we make use of K in this context we assume it to be rational. In the context of rule-based reasoning it may be of particular interest to observe that an implication is 1-satisfied exactly when the head is at least as true as the body. For a set of formulas Π, we say that a truth assignment ν is a *$K-$fuzzy model* if all formulas in Π are K-satisfied by ν.

In place of the database in the classical setting, we instead consider *(finite) partial truth assignments*, that is, partial functions $\tau : GAtoms \to (0,1]$ that are defined for a finite number of ground atoms. Let (Π, τ) be a pair where Π is a set of formulas and τ is a partial truth assignment, a *K-fuzzy model* of (Π, τ) is a K-fuzzy model ν of Π where $\nu(G) = \tau(G)$ for every ground atom G for which τ is defined. Whenever we talk about formulas Π and partial truth assignments τ we use $ADom$ to refer to their *active domain*, that is, the subset of the domain that is mentioned in either Π or $\{G \in GAtoms \mid \tau(G) \text{ is defined.}\}$. We write $GAtoms[Adom]$ to indicate $GAtoms$ restricted to groundings over $Adom$.

K-fuzzy models and their theory have been introduced and studied in the context of logic programming by Ebrahim (2001) but under different many-valued semantics based on the Gödel t-norm.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Example 1. *Consider a labelling of images as described in the introduction, represented by predicate Label. Additionally, we have a predicate corresponding to whether the image was taken in a polar region. Suppose we have the following observations for image* i_1*.*

$$\tau(\mathit{Label}(i_1, \text{Whale})) = 0.8 \qquad \tau(\mathit{PolarRegion}(i_1)) = 0.7.$$

Now we consider satisfaction of the following ground rule under the semantics studied by Ebrahim (2001) in contrast to Łukasiewicz semantics.

$$\mathit{Label}(i_1, \text{Whale}) \odot \mathit{PolarRegion}(i_1) \to \mathit{Orca}(i_1).$$

If we interpret implication and $\odot$ *under the semantics proposed by Ebrahim (2001), the body has truth* 0.7 *(the Gödel t-norm of* 0.8 *and* 0.7*). The implication evaluates to the maximum of* 0.3 *(=* 1*− truth of the body) and the truth of the head. Hence, a model that* 1*-satisfies the rule would have to satisfy* $\mathit{Orca}(i_1)$ *with truth degree* 1*. This can be unintuitive, especially in the context of logic programming, since the truth degree of the fact is not inferred from the truth of the body.*

In Łukasiewicz semantics, the body has truth degree 0.5 *following the intuition that the conjunction of two uncertain observations becomes even less certain. For the implication to be* 1*-satisfied it suffices if* $\mathit{Orca}(i_1)$ *is true to at least degree* 0.5*. We see that the inferred truth degree of the head naturally depends on the truth of the body.*

3 MV-Datalog and minimal fuzzy models

In this section, we propose the extension of Datalog to Łukasiewicz semantics for many-valued reasoning. We show that MV-Datalog allows for an analogue of minimal model semantics in the fuzzy setting. Moreover, we give a characterisation of such minimal models in terms of an optimisation problem over the ground program corresponding to the oblivious chase.

3.1 MV-Datalog

Definition 1 (MV-Datalog Program)**.** *An* MV-Datalog program Π *is a set of Ł formulas of the form*

$$B_1 \odot B_2 \odot \ldots \odot B_n \to H,$$

where all B_i*, for* $1 \leq i \leq n$*, and* H *are relational atoms.*

As noted in Section 2, we consider a partial truth assignment τ in the place of a database. We will therefore also refer to τ as the database. We call a pair Π, τ of an MV-Datalog programme and database an *MV-instance*. We can map a MV-Datalog program Π naturally to a Datalog program by substituting $\odot$ by $\wedge$. We refer to the resulting Datalog program as Π^{crisp}. For the respective crisp version of τ we write D_τ for the database containing all facts G for which $\tau(G)$ is defined and greater than 0.

Analogous to fact entailment in classical Datalog, we have the central decision problem K-Truth, whether a fact is true to at least a degree c in all models, as follows.

K-Truth

Input MV-instance (Π, τ), ground atom G, $c \in [0, 1]$
Output $\nu(G) \geq c$ for all K-fuzzy models ν of (Π, τ)?

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

MV-Datalog is a proper extension of Datalog in the sense that for $K = 1$ and when all ground atoms in the database are assigned truth 1, its models coincide exactly with those of Datalog programs.

Note that in contrast to Datalog, MV-instances do not always have models. For example, consider a program consisting only of rule $R(x) \to S(x)$. For any $K > 0$, there is a database τ that assigns truth 1 to $R(a)$ and some truth less than K to $S(a)$ such that the rule is not K-satisfied. This is a consequence of the definition of a K-fuzzy model ν of (Π, τ), which requires the truth values in ν to agree exactly with τ, for every fact for which τ is defined. In some settings it may be desirable to relax this slightly and consider K-fuzzy models ν where $\nu(G) \geq \tau(G)$ where $\tau(G)$ is defined.[1] Our semantics cover such a relaxation since it can be simulated by straightforward rewriting of the programme: for every relation symbol R that occurs in τ add rule $R(\mathbf{x}) \to R'(\mathbf{x})$, and replace all other occurrences of R in the programme by R'.

Ebrahim (2001) introduced minimal K-fuzzy models to study logic programming with the Gödel t-norm, as the intersection of all K-fuzzy models, where truth values of an intersection are taken to the minimum between the two values. We will use the following alternative (but equivalent) definition here that will be more convenient in our setting. For two truth assignment ν, μ we write $\nu \leq \mu$ when for all $G \in \mathit{GAtoms}$, it holds that $\nu(G) \leq \mu(G)$. We similarly write $\nu < \mu$ if $\nu \leq \mu$ and $\nu(G) < \mu(G)$ for at least one $G \in \mathit{GAtoms}$.

Definition 2. *Let Π, τ be an MV-instance. We say that a K-fuzzy model μ of (Π, τ) is* minimal *if for every K-fuzzy model ν of (Π, τ) it holds that $\mu \leq \nu$.*

Minimal K-fuzzy models behave very similar to minimal models in Datalog. In particular, the two following propositions state two key properties allows us to focus on a single minimal fuzzy model for the problem K-Truth.

Theorem 1. *Let Π, τ be an MV-instance. For every rational $K \in (0, 1]$, if (Π, τ) is K-satisfiable, then there exists a unique minimal K-fuzzy model for (Π, τ).*

Proof. We show that for every set of K-fuzzy models S, the infimum of S, defined as the truth assignment ν' with $\nu'(G) = \inf_{\nu \in S} \nu(G)$ for all $G \in \mathit{GAtoms}$, is also a K-fuzzy model for (Π, τ).

For any ground rule γ and $\nu \in S$, the body truth value $\nu(\mathrm{body}(\gamma))$ is given by $\max\{0, \left(\sum_{i=1}^{\ell} \nu(G_i)\right) - \ell + 1\}$ where the body of γ consists of ground atoms $G_1, \ldots, G_\ell$. Hence, since we have $\nu'(G_i) \leq \nu(G_i)$ for every $1 \leq i \leq \ell$ by construction, it also holds that $\nu'(\mathrm{body}(\gamma)) \leq \inf_{\nu \in S} \nu(\mathrm{body}(\gamma))$.

Therefore also $\nu'(\mathrm{body}(\gamma)) \leq (1 - K) + \inf_{\nu \in S} \nu(\mathrm{head}(\gamma))$, since all $\nu \in S$ are K-fuzzy models. That is, there exists a rational $r = \max\{0, \nu'(\mathrm{body}(\gamma)) - 1 + K\}$ (recall that we only consider rational K) such that r is less or equal to $\nu(\mathrm{head}(\gamma))$ for every $\nu \in S$ and $\nu'(\mathrm{body}(\gamma)) \leq (1 - K) + r$. Then, $\nu'(\mathrm{head}(\gamma)) = \inf_{\nu \in S} \nu(\mathrm{head}(\gamma)) \geq r$ and therefore also $\nu'(\mathrm{body}(\gamma)) \leq (1 - K) + \nu'(\mathrm{head}(\gamma))$, that is, ν' K-satisfies γ. We see that ν' is indeed a K-fuzzy model and clearly for every $\nu \in S$ we also have $\nu' \leq \nu$. Taking the infimum over all K-fuzzy models of (Π, τ) then yields the unique minimal K-fuzzy model for (Π, τ). □

[1] Considering $\nu(G) \leq \tau(G)$ allows the trivial model setting everything to truth 0.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Proposition 2. *Let G be a ground atom, $c \in [0,1]$, and μ the minimal K-fuzzy model of (Π, τ). Then $\mu(G) \geq c$ if and only if $\nu(G) \geq c$ for all K-fuzzy models of (Π, τ).*

Of particular importance to our main result is the fact that fuzzy models are, in a sense, only more fine-grained versions of classical models. In particular, since the classical case corresponds to every observation having maximal truth degree, fuzzy models are in a sense bounded from above by classical models.

Lemma 3. *Let (Π, τ) be an MV-instance. Let ν be a minimal K-fuzzy model of (Π, τ). Then for every $G \in GAtoms$, we have $\nu(G) > 0$ only if $\Pi^{crisp}, D_\tau \models G$.*

3.2 Characterising minimal fuzzy models

Minimal K-fuzzy models are a natural tool for query answering. In the following we show that we can in fact characterise the minimal K-fuzzy model for (Π, τ) in terms of a simple linear program $Opt^K_{\Pi,\tau}$ defined over the ground rules induced by the oblivious chase sequence[2] for Π^{crisp}, D_τ.

Let (Π, τ) be an MV-instance. Let $\Gamma = \{\gamma_1, \ldots, \gamma_m\}$ be the ground rules in $OGround(\Pi^{crisp}, D_\tau)$. Let $\mathcal{G} = \{G_1, \ldots, G_n\}$ be all of the ground atoms occurring in rules in Γ. For every $G_i \in \mathcal{G}$ we associate G_i with a variable x_i in our linear programme that intuitively will represent the truth degree of G_i. For γ_j of the form $G_{j_1}, \ldots, G_{j_\ell} \rightarrow G_{j_{head}}$ define

$$\text{Eval}(\gamma_j) := \sum_{k=1}^{\ell} (1 - x_{j_k}) + x_{j_{head}},$$

which directly expresses the satisfaction of rule γ_j, with variable x_{j_k} representing the truth of G_{j_k}. The linear programme $Opt^K_{\Pi,\tau}$ is then defined as follows

$$\begin{array}{lll} \text{minimise} & \sum_{i=1}^{n} x_i & \\ \text{subject to} & \text{Eval}(\gamma_j) \geq K & \text{for } 1 \leq j \leq m \\ & x_i = \tau(G_i) & \text{for } i \text{ where } \tau(G_i) \text{ is defined} \\ & 1 \geq x_i \geq 0 & \text{for } 1 \leq i \leq n. \end{array} \tag{1}$$

Recall, a *solution* of a linear programme is any assignment to the variables, a *feasible solution* is a solution that satisfies all constraints, and an *optimal solution* is a feasible solution with minimal value of the objective function. With this in mind we can state our main result for MV-Datalog, a characterisation of minimal K-fuzzy models in terms of optimal solutions of $Opt^K_{\Pi,\tau}$.

Theorem 4. *Let Π, τ be an MV-instance. Then $\mathbf{x}$ is an optimal solution of $Opt^K_{\Pi,\tau}$ if and only if $\nu_{\mathbf{x}}$ is a minimal K-fuzzy model of (Π, τ)*

By construction, any feasible solutions of $Opt^K_{\Pi,\tau}$ induces a K-fuzzy model $\nu_{\mathbf{x}}$ that assigns $\nu_{\mathbf{x}}(G_i) = x_i$ and 0 to all other ground atoms not in $\mathcal{G}$. Similarly, in the other direction we can observe that any K-fuzzy model ν of (Π, τ) can be used to construct

[2] While the use of the oblivious chase without existential quantification is untypical we explicitly require all rules induced by oblivious applications for our characterisation.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

a feasible solution of $Opt^K_{\Pi,\tau}$. This natural correspondence of solutions of $Opt^K_{\Pi,\tau}$ and K-fuzzy models of (Π, τ) is formalised in the following lemma.

Lemma 5. *Let Π, τ be an MV-instance. The following statements hold:*

1. *For any feasible solution $\mathbf{x}$ of $Opt^K_{\Pi,\tau}$, $\nu_{\mathbf{x}}$ is a K-fuzzy model of (Π, τ).*
2. *For any K-fuzzy model ν of (Π, τ), the solution $\mathbf{x}$ with $x_i = \nu(G_i)$ is a feasible solution of $Opt^K_{\Pi,\tau}$.*
3. *$Opt^K_{\Pi,\tau}$ is feasible if and only if (Π, τ) is K-satisfiable.*

We say that a ground rule $\gamma \in \Gamma$ is *ν-tight* if $\nu(\gamma) = K$, that is $\nu(\text{body}(\gamma)) - 1 + K = \nu(\text{head}(\gamma))$. When a rule is not tight we refer to $\nu(\text{head}(\gamma)) - (\nu(\text{body}(\gamma)) - 1 + K)$ as the *$\nu_{\mathbf{x}}$-gap* of the rule. The gap is at least 0 for any K-satisfied ground rule. We will be interested in the structural relationships between tight rules. We refer to the ground atoms in $\mathcal{G}$ for which τ is not defined as the *derived ground atoms* of $Opt^K_{\Pi,\tau}$.

Lemma 6. *Let ν be a minimal model of an MV-instance Π, τ. Let $\mathcal{G}'$ be a non-empty set of ground atoms for which ν assigns truth greater than 0. Then there exists at least one ν-tight ground rule in $OGround(\Pi^{crisp}, D_\tau)$ such that $\text{head}(\gamma) \in \mathcal{G}'$ and $\text{body}(\gamma) \cap \mathcal{G}' = \emptyset$.*

Proof of Theorem 4. Observe, that by Lemma 3 and the fact that the chase produces all atoms that can be entailed by (Π, τ), it follows that we do not have to consider solutions with atoms other than the set $\mathcal{G}$ considered in the definition of $Opt^K_{\Pi,\tau}$.

For the if direction, let μ be a minimal model of (Π, τ) with $\mu(G) = 0$ for all ground atoms $G \notin \mathcal{G}$. By Lemma 5, μ induces a feasible solution $\mathbf{x}$ of $Opt^K_{\Pi,\tau}$ where $x_i = \mu(G_i)$. Suppose this solution is not optimal and let $\mathbf{y}$ be an optimal solution of $Opt^K_{\Pi,\tau}$. Let $\mathcal{G}' \subseteq \mathcal{G}$ be the set of ground atoms G where $\nu_{\mathbf{y}}(G) < \mu(G)$. Since we assume that $\sum y_i < \sum x_i$, this set is not empty. By Lemma 6 there exists a μ-tight rule γ where the head is in $\mathcal{G}'$ but none of the body atoms of γ are. It follows that

$$\nu_{\mathbf{y}}(\text{head}(\gamma)) < \mu(\text{head}(\gamma)) = \mu(\text{body}(\gamma)) - 1 + K = \nu_{\mathbf{y}}(\text{body}(\gamma)) - 1 + K.$$

Hence, $\nu_{\mathbf{y}}$ does not K-satisfy γ. By Lemma 5 this contradicts the assumption that $\mathbf{y}$ is a feasible solution with lower objective than $\mathbf{x}$.

For the other direction, let $\mathbf{x}$ be an optimal solution of $Opt^K_{\Pi,\tau}$. Suppose now that $\nu_{\mathbf{x}}$ is not a minimal K-fuzzy model of (Π, τ). Hence, there exists some model ν' with $\nu' < \nu_{\mathbf{x}}$. Let $\mathbf{y}$ be the solution of $Opt^K_{\Pi,\tau}$ induced by ν'. Since at least one atom is less true, and none are more true in ν' we have $\sum y_i < \sum x_i$, contradicting the optimality of $\mathbf{x}$ for $Opt^K_{\Pi,\tau}$. □

Since the oblivious chase is polynomial with respect to data complexity (without existential quantification) and finding optimal solutions of linear programmes is famously polynomial (Khachiyan 1979), our characterisation of minimal K-fuzzy models also directly reveals the complexity of K-Truth. Recall that for $K = 1$, **P**-hardness is inherited from classical Datalog (see Dantsin *et al.* 2001).

Corollary 7. *Fix a rational $K \in (0, 1]$.* K-Truth *is in* **P** *with respect to data complexity. Moreover,* 1-Truth *is* **P***-complete in data complexity.*

To conclude this section we briefly note why it is important to use the oblivious chase as a basis for the ground rules that make up $Opt^K_{\Pi,\tau}$. Consider the programme $P(x), Q(x) \rightarrow$

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

$S(x)$ and $R(x) \to S(x)$. In plain Datalog, $S(a)$ can be inferred simply if either $P(a)$ and $Q(a)$, or $R(a)$ are in the database. However, in the many-valued setting the truth of $S(a)$ depends on the truth of all three facts, $P(a)$, $Q(a)$, and $R(a)$ since it needs to be at least as true as $P(x) \odot Q(a)$ and at least as true $R(a)$. Technically this means $Opt^K_{\Pi,\tau}$ needs to consider all possible paths of inferring each ground atom, and this exactly corresponds to the oblivious chase applying rules once per homomorphism from the body into the instance.

3.3 Classical behaviour for certain knowledge

We are particularly interested in settings where one wants to combine large amounts of certain knowledge, for example, classical knowledge graphs and databases, with additional fuzzy information. Going back to our example from the introduction, we have such a scenario if we want to reason on properties of the animals in the pictures by using pre-existing biological and zoological ontologies. Ideally, we want inference on the ontologies to behave exactly as in the Datalog and only differ in behaviour when such *certain knowledge* is combined with fuzzy observations and consequences of fuzzy observations. Note that particularly in the context of ontologies the question of fuzziness has also been studied extensively in the field of fuzzy description logics, see for example, Borgwardt and Peñaloza (2017), Stoilos *et al.* (2007), Lukasiewicz and Straccia (2008). Notably, the $\mathcal{EL}$ family of descrioption logics, which is closely related to rule based languages, has also been studied specifically with fuzzy semantics based on Łukasiewicz logic and shown undecidable by Borgwardt *et al.* (2017).

We show now that MV-Datalog indeed exhibits such ideal behaviour. In particular, when we consider Lemma 3 but with a database consisting only of certain knowledge in τ, that is, ground atoms for which τ evaluates to 1, we can state the following.

Theorem 8. *Let* Π*,* τ *be a MV-Datalog program and database. Let* D^1 *be the classical database containing all* $G \in GAtoms$ *where* $\tau(G) = 1$*. If* Π, τ *is* 1*-satisfiable, then for every ground atom* G*,* $\Pi^{crisp}, D^1 \models G$ *if and only if* $\mu(G) = 1$ *where* μ *is the minimal* 1*-fuzzy model of* (Π, τ)*.*

Beyond the conceptual importance of Theorem 8, the statement also has significant practical consequences. Our characterisation of minimal K-fuzzy models provides a clear algorithm for computing minimal K-fuzzy models via the oblivious chase and then solving $Opt^K_{\Pi,\tau}$. Using the oblivious chase is important for our characterisation, since different ground rules with the same head may differ in how true their body is. In classical Datalog it is of course sufficient to infer the head once, which can require significantly less effort in practice. According to Theorem 8 (and by uniqueness of minimal fuzzy models) it is not necessary to compute the truth degree of those facts that are entailed by Π^{crisp}, D^1 via $Opt^K_{\Pi,\tau}$. An implementation can use this observation to produce less ground rules (i.e., not the full oblivious chase) and smaller equivalent versions of $Opt^K_{\Pi,\tau}$ with less computational effort.

We conclude that MV-Datalog is well suited for combining large amounts of certain knowledge with fuzzy observations. Not only are the semantics on certain data equivalent to those of classical Datalog, but the computation of minimal fuzzy models also exhibits desirable properties in the presence of certain data.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

4 MV-Datalog$^{\pm}$

Existential quantification in the head of rules provide a natural way of dealing with the common problem of incomplete or missing data by declaratively stating that certain facts must exist. Such usage is especially powerful in applications where rule-based reasoning is used for complex data analysis (see e.g., Bellomarini *et al.* 2017). In the following, we introduce MV-Datalog$^{\pm}$ as an extension of MV-Datalog with a focus on providing a fuzzy reasoning language that is useful for such applications.

To obtain semantics matching our intuition for such a system, we propose an alternative to the commonly studied semantics for existential quantification for Łukasiewicz logic. We then identify certain *preferred models* that exhibit desired conceptual and computational behaviour as the basis for reasoning in MV-Datalog$^{\pm}$.

4.1 Strong existential quantification in Łukasiewicz semantics

The semantics of existential quantification in Łukasiewicz logic is commonly defined as $\nu(\exists \mathbf{x}\varphi(\mathbf{x})) = \sup\{\nu(\varphi[\mathbf{x}/\mathbf{c}]) \mid \mathbf{c} \in Dom^{|x|}\}$ where $\varphi[\mathbf{x}/\mathbf{c}]$ is the substitution of variables $\mathbf{x}$ in φ by constants $\mathbf{c}$ (cf. Hájek 1998).

However, with our focus set on practical applications, we propose alternative semantics for existential quantification that match the semantics of $\oplus$. To contrast between the aforementioned semantics of $\exists$ we refer to our semantics as *strong existential quantification*.

$$\nu(\exists \mathbf{x}\ \varphi(\mathbf{x})) = \min\{1, \sum_{\mathbf{c} \in Dom^{|x|}} \nu(\varphi[\mathbf{x}/\mathbf{c}])\}.$$

We refer to Ł extended with strong existential quantification as Ł$_{\exists}$ (following standard syntax of first order existential quantification). Such semantics for quantification in Łukasiewicz logic have also recently been studied more generally in a purely logical context by Fjellstad and Olsen (2021).

Definition 3. *A MV-Datalog$^{\pm}$ program is a set of Ł$_{\exists}$ formulas that are either MV-Datalog rules or of the form*

$$B_1 \odot B_2 \odot \cdots \odot B_n \to \exists \mathbf{x}\ \varphi(\mathbf{x}, \mathbf{y}),$$

where B_i are relational atoms, $\mathbf{y}$ are the variables in the body, and φ is formula that contains only a relational atom that uses all variables of $\mathbf{x}$.

Note that we generally do not restrict the domain in the semantics of Ł formulas. Thus, as in Datalog$^{\pm}$, interpretations are not constrained to the active domain of Π and τ but allow for the introduction of new constants. As noted above, the introduction of new constants provides a natural mechanism to deductively handle missing and incomplete data. Our proposed use of strong existential quantification is motivated by its behaviour in cases of incomplete information as illustrated in the following example.

Example 2. *We consider the following example rule, expressing that every company has a key person, to illustrate the difference between the two semantics for $\exists$.*

$$Company(y) \quad \to \exists x\ KeyPerson(x, y).$$

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Suppose we have the following database, we know for certain that Acme *is a company and we are* 0.8 *degrees confident that* Amy *is the key person of* Acme.

$$\tau(\mathit{KeyPerson}(\text{Amy}, \text{Acme})) = 0.8, \qquad \tau(\mathit{Company}(\text{Acme})) = 1.$$

We discuss the case in a 1*-fuzzy model in the following. With existential semantics via the supremum of matching ground atoms, the first rule implies the existence of some new* $\mathit{KeyPerson}(\text{N}_1, \text{Acme})$ *with truth degree* 1*. With strong existential quantification, the first rule implies only truth degree* 0.2 *for some* $\mathit{KeyPerson}(\text{N}_1, \text{Acme})$ *since the known observation* $\mathit{KeyPerson}(\text{Amy}, \text{Acme})$ *already contributes* 0.8 *degrees of truth to the head.*

Intuitively, the new constant N_1 *assumes the role of the unknown other object that possibly is the key person. In traditional semantics for* $\exists$ *we have to infer (in a* 1*-fuzzy model) that* N_1 *is certain to control* Acme*. In contrast, under strong existential semantics, the confidence in* N_1 *being the key person is determined by the known observation that* Amy *might be the key person.*

4.2 Preferred models for MV-Datalog$^{\pm}$

When considered in full generality, our proposed strong existential quantification semantics reveal a variety of theoretical challenges. To allow for K-fuzzy models ν where $\{G \in \mathit{GAtoms} \mid \nu(G) > 0\}$ is not finite would require the introduction of further technical complexity in the definition. Furthermore, it can be necessary for a model to introduce multiple new constants per ground rule to satisfy an existential quantification in certain situations as illustrated by the following example.

Example 3. *Consider database* τ *with* $\tau(S(a)) = 0.8, \tau(T(a)) = 0.2$ *and program* Π

$$\begin{aligned} S(x) \quad &\rightarrow \exists y\ P(x, y) \\ P(x, y) \quad &\rightarrow T(x). \end{aligned}$$

Then there is no 1*-fuzzy model with only one new constant. However, there is a solution where the facts* $P(a, n_1), P(a, n_2), P(a, n_3), P(a, n_4)$*, with new constants* $n_1, \ldots, n_4$*, are all assigned truth degree* 0.2*.*

Infinite models and the phenomenon illustrated by Example 3 present interesting topics for further theoretical study. For practical applications we propose to avoid these corner cases and focus on natural *preferred models* that avoid these issues and provide predictable outcomes.

Recall that the oblivious chase sequence introduces new constants exactly once for every possible true grounding of the body. This limitation on the number of new constants provides a natural balance between not overly restricting facts with null values in models while also avoiding extreme situations as in Example 3 where arbitrarily many constants are introduced to satisfy the rule for a single ground body. We also restrict minimality of fuzzy models to only those ground atoms without labelled nulls. Minimising also the truth of atoms with labelled nulls would intuitively correspond to a kind of soft closed-world assumption, where we want to infer truth of statements under the assumption that the unknown information is minimal.

Definition 4. *Let* Π, τ *be an MV*$^{\pm}$*-instance. Let* μ*,* ν *be two* K*-fuzzy models of* (Π, τ)*. We say that* $\mu \leq_a \nu$ *if for all* $G \in \mathit{GAtoms}[\mathit{Adom}]$ *it holds that* $\mu(G) \leq \nu(G)$*.*

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Definition 5. *We say that a K-fuzzy model ν of MV-Datalog$^{\pm}$ instance (Π, τ) has an* oblivious base *if $OLim(\Pi^{crisp}, D_\tau)$ is finite, if for every $G \in GAtoms$ it holds that $\nu(G) > 0$ only if $G \in OLim(\Pi^{crisp}, D_\tau)$. We say that ν is a* preferred *K-fuzzy model if it is minimal with respect to the partial ordering $\leq_a$ of all K-fuzzy models with an oblivious base.*

This definition of preferred models captures the previously outlined intuition of which kinds of models we want to consider for practical applications. That is, we propose preferred models as the desired output of an MV-Datalog$^{\pm}$ system. An MV$^{\pm}$-instance does not necessarily have a unique preferred model. The question of how to compare preferred models among each other is left as an open question.

We now extend the definition of $Opt^K_{\Pi,\tau}$ from Section 3.1 to programme $\exists\text{-}Opt^K_{\Pi,\tau}$ for an MV$^{\pm}$-instance Π, τ. Let $\mathcal{G}$ and Γ be as in the previous definition of $Opt^K_{\Pi,\tau}$. For ground rules that are the result of grounding a rule without existential quantification, Eval is defined the same as it was for $Opt^K_{\Pi,\tau}$. Otherwise, let γ be a ground rule obtained from grounding of an existential rule. Let G_h be the atom in the head of γ and let N be the labelled nulls at the positions in G_h that were existentially quantified in the original rule before grounding. We say that a ground atom G *matches* G_h if G can be obtained from G_h by replacing the labelled nulls in N by some other values of the domain. Let $G_{\beta_1}, \ldots, G_{\beta_k}$ be the ground atoms in $\mathcal{G}$ that match G_h. We can then extend our previous definition of Eval to groundings of existential rules as

$$\text{Eval}(\gamma) := \sum_{j=1}^{\ell} (1 - x_{\alpha_j}) + \sum_{i=1}^{k} x_{\beta_i}$$

with variables x_{α_j} and x_{β_i} corresponding, respectively, to the truth values of the ground atoms in the body and the head. Then $\exists\text{-}Opt^K_{\Pi,\tau}$ is the linear programme defined the same as $Opt^K_{\Pi,\tau}$, but with objective function

$$\text{minimise} \quad \textstyle\sum_{i=1}^{n} w_i x_i, \tag{2}$$

where x_i provides the truth value of G_i, $w_i = 1$ if $G_i \in GAtoms[Adom]$ and $w_i = 0$ otherwise. That is, ground atoms that contain nulls do not contribute to the objective. Recall from Example 3 that this does not imply that they can be simply set to be fully true.

Theorem 9. *Let Π, τ be a MV-Datalog program and database where $OLim(\Pi^{crisp}, D_\tau)$ is finite. The following two statements hold.*

1. *If (Π, τ) has a preferred K-fuzzy model, then $\exists\text{-}Opt^K_{\Pi,\tau}$ is feasible.*
2. *For any optimal solution $\mathbf{x}$ of $\exists\text{-}Opt^K_{\Pi,\tau}$, we have that $\nu_{\mathbf{x}}$ is a preferred K-fuzzy model for (Π, τ).*

From our definition of preferred models we are naturally interested in identifying cases where the oblivious chase is finite. There exist syntactic restrictions for which we can show that this is always the case. In particular, we can identify programmes where the oblivious chase is finite based on the notion of weakly acyclic programmes as studied by Fagin *et al.* (2005). Technical details of this question are discussed in Appendix B of the arXiv version of this paper (cf., Lanzinger *et al.* 2022).

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

5 Conclusion and outlook

We have introduced MV-Datalog and MV-Datalog$^{\pm}$, two many-valued rule-based reasoning languages following Łukasiewicz semantics. We provide theoretical foundations for MV-Datalog by characterising its minimal models in terms of an optimisation problem and showing that fuzzy fact entailment in MV-Datalog has the same complexity as fact entailment in Datalog. For MV-Datalog$^{\pm}$ we introduce alternative semantics for existential quantification in Łukasiewicz logic that better match the popular usage of $\exists$ for handling missing data. As a first starting point for the practical use of MV-Datalog$^{\pm}$ we introduce preferred models and show how to compute them.

We argue that MV-Datalog and its extensions can be effective languages for reasoning in settings with uncertain data. Therefore the development of a system based on the results given here is the natural next step to validate efficiency and effectiveness in comparison to other formalisms. This also entails the study of how the computation of minimal or preferred models can be further optimised, ideally requiring fewer ground rules than the methods described here.

Beyond practical considerations, a variety of natural theoretical questions have been left open. Prime among them are the questions of how fuzzy models of MV-Datalog behave with stratified negation, and how to handle the behaviour of MV-Datalog$^{\pm}$ instances with an infinite oblivious chase. The latter is particularly intriguing since truth degrees effectively decreases monotonically along paths of derivation. We therefore suspect there to be relevant classes of problems where the oblivious chase is infinite but instances always permit finite fuzzy models.

Finally, there is an intriguing possibility of combining MV-Datalog semantics with the probabilistic semantics of Problog to reason about uncertainty and likelihood as two orthogonal concepts at the same time.

Supplementary material

To view supplementary material for this article, please visit http://doi.org/10.1017/S1471068422000199.

References

Achs, Á. and Kiss, A. 1995. Fuzzy extension of datalog. *Acta Cybernetica 12*, 2, 153–166.

Bach, S. H., Broecheler, M., Huang, B. and Getoor, L. 2017. Hinge-Loss Markov random fields and probabilistic soft logic. *Journal of Machine Learning Research 18*, 109:1–109:67.

Bellomarini, L., Gottlob, G., Pieris, A. and Sallinger, E. 2017. Swift logic for big data and knowledge graphs. In *Proceedings of IJCAI*. ijcai.org, 2–10.

Bellomarini, L., Laurenza, E., Sallinger, E. and Sherkhonov, E. 2020. Reasoning under uncertainty in knowledge graphs. In *RuleML+RR*. Springer, 131–139.

Bobillo, F. and Straccia, U. 2016. The fuzzy ontology reasoner fuzzydl. *Knowledge-Based Systems 95*, 12–34.

Borgwardt, S., Cerami, M. and Peñaloza, R. 2017. Łukasiewicz fuzzy EL is undecidable. In *Proceedings of DL*. CEUR Workshop Proceedings, vol. 1879. CEUR-WS.org.

Borgwardt, S. and Peñaloza, R. 2017. Fuzzy description logics - A survey. In *Proceedings of SUM*. Lecture Notes in Computer Science, vol. 10564. Springer, 31–45.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press

Calì, A., Gottlob, G. and Kifer, M. 2013. Taming the infinite chase: Query answering under expressive relational constraints. *JAIR 48*, 115–174.

Calì, A., Gottlob, G. and Lukasiewicz, T. 2012. A general Datalog-based framework for tractable query answering over ontologies. *Journal of Web Semantics 14*, 57–83.

Cavallo, R. and Pittarelli, M. 1987. The theory of probabilistic databases. In *VLDB'87*. Morgan Kaufmann, 71–81.

Dantsin, E., Eiter, T., Gottlob, G. and Voronkov, A. 2001. Complexity and expressive power of logic programming. *ACM Computing Surveys 33*, 3, 374–425.

Ebrahim, R. 2001. Fuzzy logic programming. *Fuzzy Sets and Systems 117*, 2, 215–230.

Fagin, R., Kolaitis, P. G., Miller, R. J. and Popa, L. 2005. Data exchange: Semantics and query answering. *Theoretical Computer Science 336*, 1, 89–124.

Fjellstad, A. and Olsen, J.-F. 2021. IKT$^{\omega}$ and Łukasiewicz-Models. *Notre Dame Journal of Formal Logic 62*, 2, 247–256.

Hájek, P. 1998. *Metamathematics of Fuzzy Logic*. Trends in Logic, vol. 4. Kluwer.

Iranzo, P. J. and Sáenz-Pérez, F. 2018. A fuzzy datalog deductive database system. *IEEE Transactions on Fuzzy Systems 26*, 5, 2634–2648.

Khachiyan, L. G. 1979. A polynomial algorithm in linear programming. In *Doklady Akademii Nauk*, vol. 244. Russian Academy of Sciences, 1093–1096.

Lanzinger, M., Sferrazza, S. and Gottlob, G. 2022. Mv-datalog+-: Effective rule-based reasoning with uncertain observations. CoRR, abs/2202.01718.

Lee, H. S., Jung, H., Agarwal, A. A. and Kim, J. 2017. Can deep neural networks match the related objects?: A survey on imagenet-trained classification models. CoRR, abs/1709.03806.

Lukasiewicz, T. and Straccia, U. 2008. Managing uncertainty and vagueness in description logics for the semantic web. *Journal of Web Semantics 6*, 4, 291–308.

Maier, D., Tekle, K. T., Kifer, M. and Warren, D. S. 2018. Datalog: Concepts, history, and outlook. In *Declarative Logic Programming: Theory, Systems, and Applications*. ACM/Morgan & Claypool.

Mishkin, D., Sergievskiy, N. and Matas, J. 2017. Systematic evaluation of convolution neural network advances on the imagenet. *Computer Vision and Image Understanding 161*, 11–19.

Preining, N. 2010. Gödel logics - A survey. In *Proceedings of LPAR*. Lecture Notes in Computer Science, vol. 6397. Springer, 30–51.

Raedt, L. D., Kimmig, A. and Toivonen, H. 2007. Problog: A probabilistic prolog and its application in link discovery. In *Proceedings of IJCAI*, 2462–2467.

Richardson, M. and Domingos, P. M. 2006. Markov logic networks. *Machine Learning 62*, 1-2, 107–136.

Stoilos, G., Stamou, G. B., Pan, J. Z., Tzouvaras, V. and Horrocks, I. 2007. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research 30*, 273–320.

Suciu, D., Olteanu, D., Ré, C. and Koch, C. 2011. *Probabilistic Databases*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers.

Vojtás, P. 2001. Fuzzy logic programming. *Fuzzy Sets and Systems 124*, 3, 361–370.

https://doi.org/10.1017/S1471068422000199 Published online by Cambridge University Press