

On Interval Dynamic Logic

Regivan H. N. Santiago¹, Benjamín Bedregal¹, Alexandre Madeira² and
Manuel A. Martins³

¹ Univ. Federal do Rio Grande do Norte — UFRN
Dep. de Informática e Matemática Aplicada — DIMAp
Group for Logic, Language, Information, Theory and Applications — LoLITA,
Natal-RN-Brazil

² HASLab, INESC-TEC, Dep. Informática Univ. Minho, Portugal

³ CIDMA - Dep. Mathematics, Univ. Aveiro, Portugal

Abstract. The wide number of languages and programming paradigms, as well as the heterogeneity of ‘programs’ and ‘executions’ require new generalisations of propositional dynamic logic. The dynamisation method, introduced in [20], contributed on this direction with a systematic parametric way to construct Multi-valued Dynamic Logics able to handle systems where the uncertainty is a prime concern. The instantiation of this method with the Lukasiewicz arithmetic lattice over $[0, 1]$, that we derive here, supports a general setting to design and to (fuzzy-) reason about systems with uncertainty degrees in their transitions.

For the verification of real systems, however, there are no *de facto* methods to accommodate exact truth degrees or weights. Instead, the traditional approach within scientific community is to use different kinds of approximation techniques.

Following this line, the current paper presents a framework where the representation values are given by means of intervals. Technically this is achieved by considering an ‘interval version’ of the Kleene algebra based on the $[0, 1]$ Lukasiewicz lattice. We also discuss the ‘intervalisation’ of \mathbb{L} action lattice (in the lines reported in [28]) and how this class of algebras behaves as an (interval) semantics of multi-valued dynamic logic.

1 Introduction

Dynamic Logics (DL) are extensions of modal logic. They are recognised as the most adequate logics to reason about computational systems in an assertional way [12]. In its origin, DL was introduced by V. Pratt [26] as a modal logic suitable to represent and reason about Hoare triples. Since then, DL assumed a central role in the programs verification. Today, not only in response of the explosion of programming and specification languages, but also in the emerging heterogeneous nature that a program can assume, a wide family of DL were defined to be applied to more general complex behaviors. This ranges from the standard versions for sequential imperative programmes (e.g. [12]) to other versions tailored for new computing paradigms, either probabilistic systems, following the original work of D. Kozen [17], hybrid systems with the

differential dynamic logic by A. Platzer [25] or even quantum versions due to A. Baltag and S. Smets [1]. Within this variety of dynamic logics, in [20], it is studied a method for a systematic construction of many-valued dynamic logics. The method is parametrised by an action lattice that supports both the computational paradigm and the truth space, combining in just one structure the underlying Kleene algebra for the computations and residuated lattices for the proposition assertions.

On the other hand, logics with many-valued semantics are applied in a variety of fields like Decision Making, Image Processing, Clustering, etc. One of such logics, which is a very important logic, is that of Łukasiewicz [8], which semantics is based on the residuated lattice $\mathbb{L} = \langle [0, 1], \rightarrow, 0, 1 \rangle$ — where $a \rightarrow b = \min(1, 1 - a + b)$. In this logic the truth-values may be thought of as arising from normalized measurements of bounded physical observables, just as boolean truth-values arise from yes/no-observables [24, §1.6]. In this context, it is worth to look into the instantiation of the ‘dynamisation’ parametrised by the Łukasiewicz action lattice $\mathbb{L} = \langle [0, 1], \max, \odot, 0, 1, *, \rightarrow, \min \rangle$. The space of values $[0, 1]$ models exact measurements which is far from the real-world. In fact, any measurement presupposes an uncertainty which is not encoded by the elements of $[0, 1]$. For instance, there is no machine representation of irrational numbers. This justifies the use of approximations to those practices. In order to capture and deal with such uncertainty the indicated approach is to **represent** [4] the space $[0, 1]$ by the space of closed intervals of its elements $\mathbb{I}([0, 1]) = \{[a, b] \mid [a, b] \subseteq [0, 1]\}$. Representation here means correction in the sense of interval mathematics [13, 28] and can be summarised by the following expressions:

If $a \in A$ and $b \in B$, then:

1. $a^* \in A^*$, $a \rightarrow b \in A \rightarrow B$,
2. $\max(a, b) \in \max(A, B)$, $\min(a, b) \in \min(A, B)$ and
3. $a \odot b \in A \odot B$

Many interesting questions arise in the interval setting. Since $\mathbb{I}([0, 1])$ extends $[0, 1]$, what are the action lattices properties which are maintained or destroyed? Actually, we proved that the obtained interval structure is a Kleene Algebra (see Theorem 1). This paper’s contribution can be useful on the specification and analysis of programs involving uncertainty degrees in the execution (transitions). Moreover, based on the Conway matrix constructions [9], we have a support for the composition under the Kleene operations (sequential composition, choice and *-closure) of transition systems weighed by intervals.

However, the notion of action lattice is not enough to abstract the resulting structure. Namely, the residuum property does not hold (see Proposition 3). Although, we still can derive a Multi-valued dynamic logic from such structure such that the semantics behaves as expected: the value of a sentence in the standard case belongs to the respective interval interpretation.

Outline of the paper. The paper is structured as follows: Section 2 introduces a Multi-valued Dynamic Logic built on the \mathbb{L} -action lattice, following the ‘dynamisation’ method of [20]. Then, Section 3 makes an overview on the interval

arithmetics and applies it to the ‘intervalisation’ of \mathbf{L} . It is also proved that this structure is, in fact, a Kleene algebra. In Section 4, we discuss the properties which are preserved and lost in this procedure. Finally, in Section 5 we present some direction for future work.

2 An \mathbf{L} -Fuzzy Dynamic Logic

Many-valued versions of Modal Logics have been discussed in the literature along the times; the purposed logics vary in the focus where the many-valuedness is presented: in accessibility relations, in propositions interpretation or in both. The latter is the case of the works [10, 11] of M. Fitting suggest a logic with many-valuedness evaluated in finite Heyting algebras. Later it was deeply investigated by F. Bou et al in [6], who adopted the more generic truth support of finite integral commutative residuated lattices.

The literature is not so rich at respect of Many-Valued Dynamic Logics. J. Hughes et al introduced in [14] a propositional dynamic logic over the continuum truth $(0, 1)$ -lattice with the standard fuzzy residues. However, from the perspective of dynamic logic, this formalism is quite restrictive, since it leaves behind both transitive closure and non deterministic choice. In the context of rational decision theory, C. Liao [19] introduced also another different many-valued dynamic logic w.r.t. the specific continuum truth $(0, 1)$ -lattice.

A systematic method to build Multi-valued Dynamic Logics was then introduced in [20, 21]. This method is parametrized by an action lattice [16], an algebraic structure that provides a generic support for computational space (as a Kleene algebra) and for truth space (as residuated lattice). The logic introduced in this section is based on this work and can be captured as an instantiation of this method.

2.1 The Łukasiewicz action lattice

The role of the Łukasiewicz residuated lattice, i.e., the algebraic structure

$$\mathbf{L} = ([0, 1], \max, \odot, 0, 1, \rightarrow)$$

with

- $x \rightarrow y = \min(1, 1 - x + y)$ and
- $x \odot y = \max(0, y + x - 1)$,

is taken as the standard fuzzy truth space [30]. Moreover, as stated above, we are looking for a structure suitable to support a fuzzy computational model. Whenever the \max and the \odot operators are used to model the choice and composition of atomic actions, we need to consider a Kleene operator to model the recursive iteration of programs. These constitute the components of an action lattice [16], the structure taken in [20, 21] as a generic parameter for a multi-valued logic definition. More precisely:

Definition 1. An action lattice is a tuple

$$\mathbf{A} = (A, +, ;, 0, 1, *, \rightarrow, \cdot)$$

where A is a set, 0 and 1 are constants, $*$ is an unary operation in A and $+, ;, \rightarrow$ and \cdot are binary operations in A satisfying the axioms enumerated in Figure 1, where the relation \leq is induced by $+$: $a \leq b$ iff $a + b = b$.

$$\begin{array}{ll}
a + (b + c) = (a + b) + c & (1) \\
a + b = b + a & (2) \\
a + a = a & (3) \\
a + 0 = 0 + a = a & (4) \\
a; (b; c) = (a; b); c & (5) \\
a; 1 = 1; a = a & (6) \\
a; (b + c) = (a; b) + (a; c) & (7) \\
(a + b); c = (a; c) + (b; c) & (8) \\
a; 0 = 0; a = 0 & (9) \\
1 + a + (a^*; a^*) \leq a^* & (10) \\
a; x \leq x \Rightarrow a^*; x \leq x & (11) \\
x; a \leq x \Rightarrow x; a^* \leq x & (12) \\
a; x \leq b \Leftrightarrow x \leq a \rightarrow b & (13) \\
a \rightarrow b \leq a \rightarrow (b + c) & (14) \\
(x \rightarrow x)^* = x \rightarrow x & (15) \\
a \cdot (b \cdot c) = (a \cdot b) \cdot c & (16) \\
a \cdot b = b \cdot a & (17) \\
a \cdot a = a & (18) \\
a + (a \cdot b) = a & (19) \\
a \cdot (a + b) = a & (20)
\end{array}$$

Fig. 1. Axiomatisation of action lattices (from [16])

Note that, by (19) and (20), the natural order \leq can be equivalently defined by $a \leq b$ iff $a \cdot b = a$. Observe that by restricting the definition of \mathbf{A} to the structure $(A, +, ;, 0, 1, *)$ axiomatised by (1)–(12) we obtain the definition of a Kleene algebra [9, 18]. In the context of this work, this will be called the underlying Kleene algebra of \mathcal{A} . Moreover, by considering structure $(A, +, ;, 0, 1, \rightarrow, *)$ axiomatised by (1)–(15) we obtain the definition of (left-residuated) action algebra [27].

For the illustration of the structure with several examples and properties we suggest [20]. Just as example, we can consider a discrete 3-valued lattice underling the 3-valued logic:

Example 1 (3 - linear three-value lattice). The explicit introduction of a denotation for unknown gives rise to the the following three elements linear lattice

$$\mathbf{3} = (\{\top, u, \perp\}, \vee, \wedge, \perp, \top, *, \rightarrow, \wedge)$$

where

$$\begin{array}{c|ccc}
\vee & \perp & u & \top \\
\hline
\perp & \perp & u & \top \\
u & u & u & \top \\
\top & \top & \top & \top \\
\hline
\wedge & \perp & u & \top \\
\hline
\perp & \perp & \perp & \perp \\
u & \perp & u & u \\
\top & \perp & u & \top \\
\hline
\rightarrow & \perp & u & \top \\
\hline
\perp & \top & \top & \top \\
u & \perp & \top & \top \\
\top & \perp & u & \top \\
\hline
* & \perp & \top & \top \\
\hline
\perp & \perp & \top & \top \\
u & \perp & \top & \top \\
\top & \perp & u & \top \\
\hline
\end{array}$$

It is easy to observe that, as a consequence of axiom (10), whenever $\top = 1$, we have that $x^* = 1$, for all x . Hence we have all the ingredients to introduced the Łukasiewicz arithmetic lattice, a structure that plays a main role in the theory developed in the sequel:

Definition 2 (Ł - the Łukasiewicz arithmetic lattice). *The Łukasiewicz arithmetic lattice is the structure*

$$\mathbf{L} = ([0, 1], \max, \odot, 0, 1, *, \rightarrow, \min)$$

where

- $x \rightarrow y = \min(1, 1 - x + y)$,
- $x \odot y = \max(0, y + x - 1)$ and
- $x^* = 1$.

2.2 The Ł-Fuzzy Dynamic Logic

Signatures of \mathbf{LDL} are exactly the same of the ones of Propositional Dynamic Logic: signatures are pairs (Π, Prop) of disjoint sets of atomic programs Π and of propositions symbols Prop .

Formulas of \mathbf{LDL} consists in the positive fragment of Propositional Dynamic Logic: the set of Π -programs, denoted by $\text{Prg}(\Pi)$, consists of all expressions generated by

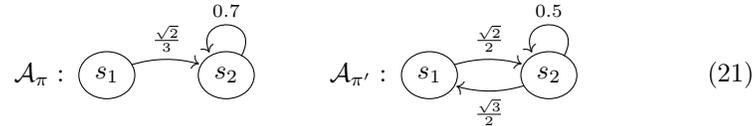
$$\pi \ni \pi_0 \mid \pi; \pi \mid \pi + \pi \mid \pi^*$$

for $\pi_0 \in \Pi$. Given a signature (Π, Prop) , we define the \mathbf{LDL} -formulae for (Π, Prop) , denoted by $\text{Fm}^{\mathbf{LDL}}(\Pi, \text{Prop})$, as the ones generated by the grammar

$$\rho \ni \top \mid \perp \mid p \mid \rho \vee \rho \mid \rho \wedge \rho \mid \rho \rightarrow \rho \mid \rho \leftrightarrow \rho \mid \langle \pi \rangle \rho \mid [\pi] \rho$$

for $p \in \text{Prop}$ and $\pi \in \text{Prg}(\Pi)$.

Semantics. As expectable, the interpretation of atomic programs are Kripke structures with weighted transitions. For instance, atomic programs $\Pi = \{\pi, \pi'\}$ can be realized by the structures



where that tags mention the uncertainty level of each states transitions. These weighted transition systems are usually represented by the underlying adjacency matrices

$$\mathcal{A}_\pi = \begin{bmatrix} 0 & \frac{\sqrt{2}}{3} \\ 0 & 0.7 \end{bmatrix} \quad \mathcal{A}_{\pi'} = \begin{bmatrix} 0 & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{3}}{2} & 0.5 \end{bmatrix}$$

Moreover, we need a mathematical framework to interpret composed programs, i.e., regular expressions of atomic programs. In other words, we need to consider a computational space for LDL where the programs are interpreted. Based on the classic matricial constructions over Kleene algebras (see [9, 18]) we consider the structure

$$\mathbb{M}_n(\mathbf{L}) = (M_n(\mathbf{L}), \max, \odot, 0, 1, *) \quad (22)$$

as follows:

1. $M_n(\mathbf{L})$ is the space of $(n \times n)$ -matrices over \mathbf{L}
2. for any $A, B \in M_n(\mathbf{L})$, define $M = \max(A, B)$ by $M_{ij} = \max(A_{ij}, B_{ij})$, $i, j \leq n$.
3. for any $A, B \in M_n(\mathbf{L})$, define $M = A \odot B$ by taking

$$\max(A_{i1} \odot B_{1j}, \max(A_{i2} \odot B_{2j}, (\dots, \max(A_{in} \odot B_{nj}) \dots)))$$

4. the matricial 1 and 0 are the $(n \times n)$ -matrices defined by $1_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$

and $0_{i,j} = 0$, for any $i, j \leq n$.

5. for any $M = [a] \in \mathbb{M}_1(\mathbf{A})$, $M^* = [a^*]$;

for any $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \in M_n(\mathbf{A})$, $n > 1$, where A and D are square matrices, define

$$M^* = \left[\begin{array}{c|c} F^* & F^* \odot (B \odot D^*) \\ \hline (D^* \odot C) \odot F^* & \max(D^*, (D^* \odot (C \odot (F^* \odot (B \odot D^*)))))) \end{array} \right]$$

where $F = \max(A, B \odot (D^* \odot C))$. Note that this construction is recursively defined from the base case (where $n = 2$) where the operations of the base action lattice \mathbf{A} are used.

A classic result (e.g. [9, 18]) establishes that Kleene algebras are closed under formation of matrices. This justifies the adoption of $\mathbb{M}_n(\mathbf{L})$ as a well behaved computational space for LDL .

Theorem 1. *The structure $\mathbb{M}_n(\mathbf{L}) = (M_n(\mathbf{L}), \max, \odot, 0, 1, *)$ defined above is a Kleene algebra.*

LDL -models for a set of propositions Prop and programs Π , denoted by $\text{Mod}^{\text{LDL}}(\Pi, \text{Prop})$, consists of tuples

$$\mathcal{A} = (W, V, (\mathcal{A}_\pi)_{\pi \in \Pi})$$

where W is a finite set (of states), $V : \text{Prop} \times W \rightarrow [0, 1]$ is a function, and $\mathcal{A}_\pi \in \mathbb{M}_n(\mathbf{L})$, with n standing for the cardinality of W .

The interpretation of programs in these models belongs to the space of the matrices over the underlying Kleene algebra of \mathbf{L} . Each matrix represents the

effect of a program executing from any point of the model. Formally, the interpretation of a program $\pi \in \text{Prg}(\Pi)$ in a model $\mathcal{A} \in \text{Mod}^{\mathbf{L}\mathcal{D}\mathcal{L}}(\Pi, \text{Prop})$ is recursively defined, from the set of atomic programs $(\mathcal{A}_\pi)_{\pi \in \Pi}$, as follows:

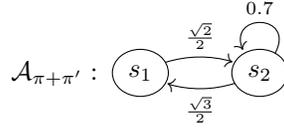
$$\mathcal{A}_{\pi;\pi'} = \mathcal{A}_\pi \odot \mathcal{A}_{\pi'}, \mathcal{A}_{\pi+\pi'} = \max(\mathcal{A}_\pi, \mathcal{A}_{\pi'}) \text{ and } \mathcal{A}_{\pi^*} = \mathcal{A}_\pi^*.$$

together with the constants interpretations $\mathcal{A}_1 = \mathbf{1}$ and $\mathcal{A}_0 = \mathbf{0}$.

returning to our running example, we are able to calculate the interpretation of the program $\mathcal{A}_{\pi+\pi'}$ by making

$$\mathcal{A}_{\pi+\pi'} = \max(\mathcal{A}_\pi, \mathcal{A}_{\pi'}) = \max\left(\begin{bmatrix} 0 & \frac{\sqrt{2}}{3} \\ 0 & 0.7 \end{bmatrix}, \begin{bmatrix} 0 & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{3}}{2} & 0.5 \end{bmatrix}\right) = \begin{bmatrix} 0 & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{3}}{2} & 0.7 \end{bmatrix} \quad (23)$$

that represents the following weighted transition system:



By considering the interpretation of the propositions $\text{Prop} = \{p, q\}$ as $V(p, s_1) = 0.1$, $V(q, s_1) = 0.5$, $V(p, s_2) = \frac{\pi}{4}$ and $V(q, s_2) = 0.75$ we have a complete description of a concrete $(\{\pi, \pi'\}, \{p, q\})$ -model $\mathcal{A} = (\{s_1, s_2\}, V, (\mathcal{A}_p)_{p \in \{\pi, \pi'\}})$.

Satisfaction.

As mentioned above, the carrier of \mathbf{L} corresponds to the space of truth degrees for $\mathbf{L}\mathcal{D}\mathcal{L}$. Hence, the graded satisfaction relation for a model $\mathcal{A} \in \text{Mod}^{\mathbf{L}\mathcal{D}\mathcal{L}}(\Pi, \text{Prop})$, consists of a function

$$\models : W \times \text{Fm}^{\mathbf{L}\mathcal{D}\mathcal{L}}(\Pi, \text{Prop}) \rightarrow \mathbf{L}$$

recursively defined as follows:

- $(w \models \top) = 1$
- $(w \models \perp) = 0$
- $(w \models p) = V(p, w)$, for any $p \in \text{Prop}$
- $(w \models \rho \wedge \rho') = \min \{(w \models \rho), (w \models \rho')\}$
- $(w \models \rho \vee \rho') = \max \{(w \models \rho), (w \models \rho')\}$
- $(w \models \rho \rightarrow \rho') = (w \models \rho) \rightarrow (w \models \rho')$
- $(w \models \langle \pi \rangle \rho) = \max \{ \mathcal{A}_\pi(w, w') \odot (w' \models \rho) \mid w' \in W \}$
- $(w \models [\pi] \rho) = \min \{ \mathcal{A}_\pi(w, w') \rightarrow (w' \models \rho) \mid w' \in W \}$

In order to illustrate the definition, the calculation of the truth degree of the formula $\langle \pi + \pi' \rangle (p \rightarrow q)$ in the introduced model \mathcal{A} can be achieved as follows:

$$\begin{aligned}
(s_1 \models \langle \pi + \pi' \rangle (p \rightarrow q)) &= \max(0 \odot (0.1 \rightarrow 0.5), \frac{\sqrt{2}}{2} \odot (0.75 \rightarrow \frac{\pi}{4})) \\
&= \frac{\sqrt{2}}{2} \odot (0.75 \rightarrow \frac{\pi}{4}) \\
&= \frac{\sqrt{2}}{2} \odot \min(1, 1 - 0.75 + \frac{\pi}{4}) \\
&= \frac{\sqrt{2}}{2}
\end{aligned}$$

Therefore, we conclude with a degree of certainty $\frac{\sqrt{2}}{2}$ that, after executing $\pi + \pi'$ from the state s_1 , we have $p \rightarrow q$.

3 L-Interval algebra

As stated before, the space of values $[0, 1]$ models exact measurements/truth values which is far from the real-world. In fact, any measurement presupposes an uncertainty which is not encoded by the elements of $[0, 1]$. Another situation arises whenever an expert is unable to supply an exact membership of an object in a fuzzy set, in this case he can provide a closed subinterval of $[0, 1]$ as an expression of his inability to supply an exact answer [7]. Such closed subintervals can also be used as the resulting abstraction of the exact values provided by various experts about the same membership. Therefore, assuming the **Lukasiewicz arithmetic lattice** $\mathbf{L} = \langle [0, 1], \max, \odot, 0, 1, *, \rightarrow, \min \rangle$ as a natural space of measurements/truth values [24, §1.6] it is reasonable to investigate its interval counterpart. But what would be such interval counterpart? Before we proceed to answer this question, let's expose a little about the interval counterpart of real numbers algebra: $\langle \mathbb{R}; +, -, /, \times, 1, 0 \rangle$.

In the 50's Ramon Moore [22, 23] and Teruo Sunaga [29] proposed the so called **interval arithmetics**. Interval arithmetics is a set of operations on the set of all closed intervals $[a, b] \subseteq \mathbb{R}$. They defined the arithmetic in the following way:

1. $[a, b] + [c, d] = [a + c, b + d]$
2. $-[c, d] = [-d, -c]$
3. $[a, b] \cdot [c, d] = [\min P, \max P]$ — where $P = \{a \cdot c, a \cdot d, b \cdot c, b \cdot d\}$
4. $[a, b]^{-1} = [1/b, 1/a]$; provided that $0 \notin [a, b]$
5. $[a, b] - [c, d] = [a - d, b - c]$
6. $[a, b]/[c, d] = [a, b] \cdot ([c, d]^{-1})$

Observe what happens with each operation:

1. If $x \in [a, b]$ and $y \in [c, d]$, then $(x + y) \in [a, b] + [c, d]$,
2. If $x \in [a, b]$ and $y \in [c, d]$, then $(x \cdot y) \in [a, b] \cdot [c, d]$,
3. If $x \in [a, b]$ and $y \in [c, d]$, then $(x/y) \in [a, b]/[c, d]$, and
4. If $x \in [a, b]$, then $(-x) \in -[a, b]$.

The arithmetic on intervals reveals two desired properties: (a) **Correctness** and (b) **Optimality**.

“*Correctness*. . . when an expression is evaluated using intervals, it yields an interval containing all results of pointwise evaluations based on point values that are elements of the argument intervals.

...

Optimality. By optimality, we mean that the computed floating-point interval is not wider than necessary.”

Hickey *et.al*[13, p.1040]

The term *Correctness* connects n -ary interval operations F with n -ary real operations f and means that if F is *correct with respect to f* , then we can enfold any exact value $r \in \mathbb{R}$ in a closed interval $[a, b]$, such that $r \in [a, b]$, and then simply operate with such “envelopes” by using F , because the resulting interval $F([a, b])$ will enfold the desired result $f(r)$. Formally a function F is correct with respect to a real function f whenever:

$$r \in [a, b] \Rightarrow f(r) \in F([a, b]) \tag{24}$$

In practice, exact values are replaced by intervals which are operated with correct interval functions. Intervals enfold the exact values and provide a measure of impreciseness through its width.

Santiago *et. al* [2, 28] investigated the notion of Correctness. Instead of correctness the authors used the term **representation**, since interval expressions could be faced not just as machine representations of an exact calculation, but also as an instance of a “mathematical representation of real numbers”⁴. Beyond correctness these interval operations are also optimum; namely the resulting intervals contain only the values of real operations. We could say that the proposed algebra of intervals is the **best interval representation for the arithmetic of real numbers**.

One side-effect of this process of intervalization is the loss of algebraic properties. The resulting structure is not an Euclidean field; for example $X - X$ is not always equal to $[0, 0]$. In this paper we will also lose some properties of \mathbb{L} .

3.1 On the Interval Lukasiewicz Lattice

The Lukasiewicz arithmetic lattice $\mathbb{L} = \langle [0, 1], \max, \odot, 0, 1, *, \rightarrow, \min \rangle$ contains non-finitely representable elements; e.g. irrational numbers. In a similar way we can think of an interval algebra for \mathbb{L} . A piece of such algebra was introduced by Bedregal and Santiago in [4]. There, the authors proposed a correct interval implication for “ \rightarrow ”. In what follows we propose the interval counterpart for $\mathbb{L} = \langle [0, 1], \max, \odot, 0, 1, *, \rightarrow, \min \rangle$ in such a way that the resulting operations are correct and optimal, i.e. they are **best interval representations**.

Definition 3. Consider the real unit interval $U = [0, 1] \subseteq \mathbb{R}$ and the set $\mathbb{U} = \{[a, b] \mid 0 \leq a \leq b \leq 1\}$ of subintervals of U . For any interval $X \in \mathbb{U}$, \underline{X} is the

⁴ This idea is confirmed in some Representation Theorems of Euclidean continuous functions.

minimum of X and \overline{X} the maximum of X ; i.e. $X = [\underline{X}, \overline{X}]$. Given two intervals $X, Y \in \mathbb{U}$, let be the following partial orders on \mathbb{U} :

(i) *The product or Kulisch-Miranker order :*

$$X \leq Y \Leftrightarrow \underline{X} \leq \underline{Y} \wedge \overline{X} \leq \overline{Y}; \quad (25)$$

(ii) *The set inclusion order: for all $X, Y \in \mathbb{U}$,*

$$X \subseteq Y \Leftrightarrow \underline{Y} \leq \underline{X} \wedge \overline{X} \leq \overline{Y}. \quad (26)$$

Definition 4 ([28]). An interval $X \in \mathbb{U}$ is a **representation** of any real number $\alpha \in X$. Considering two interval representations X and Y for a real number α , X is said to be an interval representation of α better than Y , if $X \subseteq Y$. This notion can also be naturally extended for n -tuples of intervals. A function $F : \mathbb{U}^n \rightarrow \mathbb{U}$ is said to be an **interval representation** of a real function $f : U^n \rightarrow U$ if, for each $\mathbf{X} \in \mathbb{U}^n$ and $\mathbf{x} \in \mathbf{X}$, $f(\mathbf{x}) \in F(\mathbf{X})$. F is also said to be correct with respect to f . An interval function $F : \mathbb{U}^n \rightarrow \mathbb{U}$ is said to be an interval representation of a real function $f : U^n \rightarrow U$ better than an interval function $G : \mathbb{U}^n \rightarrow \mathbb{U}$, if $F(\mathbf{X}) \subseteq G(\mathbf{X})$, for each $\mathbf{X} \in \mathbb{U}^n$. The **best interval representation** of a real function $f : U^n \rightarrow U$ is the interval function $\widehat{f} : \mathbb{U}^n \rightarrow \mathbb{U}$, defined by

$$\widehat{f}(\mathbf{X}) = [\inf\{f(\mathbf{x}) \mid \mathbf{x} \in \mathbf{X}\}, \sup\{f(\mathbf{x}) \mid \mathbf{x} \in \mathbf{X}\}]. \quad (27)$$

In what follows we show the best interval representation for the Łukasiewicz arithmetic lattice $\mathbf{L} = \langle [0, 1], \max, \odot, 0, 1, *, \rightarrow, \min \rangle$. Almost all of the resulting interval representations comes from previous works. Before we go further it is noteworthy that **the following resulting structure is the best possible interval structure to represent the Łukasiewicz arithmetic lattice.**

Definition 5. Given $X, Y \in \mathbb{U}$.

1. $Max(X, Y) = [\max(\underline{X}, \underline{Y}), \max(\overline{X}, \overline{Y})]$
2. $Min(X, Y) = [\min(\underline{X}, \underline{Y}), \min(\overline{X}, \overline{Y})]$
3. $X \odot Y = [(\underline{X} \odot \underline{Y}), (\overline{X} \odot \overline{Y})] = [\max(0, \underline{X} + \underline{Y} - 1), \max(0, \overline{X} + \overline{Y} - 1)]$
4. $X \Rightarrow Y = [(\overline{X} \rightarrow \underline{Y}), (\underline{X} \rightarrow \overline{Y})] = [\min(1, 1 - \overline{X} + \underline{Y}), \min(1, 1 - \underline{X} + \overline{Y})]$
5. $X^{\otimes} = [\underline{X}^*, \overline{X}^*] = [1, 1]$.

Proposition 1. All of these interval operations are the best interval representations of the operations in \mathbf{L} ; i.e. $Max = \widehat{\max}$, $Min = \widehat{\min}$, $\odot = \widehat{\odot}$, $\Rightarrow = \widehat{\Rightarrow}$, and $\otimes = \widehat{*}$.

Proof. The operations \min and \odot are T-norms on $[0, 1]$ and the interval representations of T-norms, according to [5, Theorem 4.3], is given by $\widehat{T}(X, Y) = [T(\underline{X}, \underline{Y}), T(\overline{X}, \overline{Y})]$. The \max operation is a T-conorm on $[0, 1]$ and the interval representations of T-conorms, according to [3, Theorem 5.2], is given by $\widehat{S}(X, Y) = [S(\underline{X}, \underline{Y}), S(\overline{X}, \overline{Y})]$. According to [4, Proposition 4.4] $\Rightarrow = \widehat{\Rightarrow}$. Finally, it is trivial that $\otimes = \widehat{*}$.

Proposition 2. $X \leq Y$ iff $Max(X, Y) = Y$.

Proof. $X \leq Y$ iff $\underline{X} \leq \underline{Y}$ and $\overline{X} \leq \overline{Y}$ iff $\max(\underline{X}, \underline{Y}) = \underline{Y}$ and $\max(\overline{X}, \overline{Y}) = \overline{Y}$ iff $Max(X, Y) = Y$.

The structure $\langle \mathbb{U}, Max, Min, [0, 0], [1, 1] \rangle$ is a bounded lattice.

Theorem 2. The structure $K(\widehat{\mathbf{L}}) = \langle \mathbb{U}, \widehat{\max}, \widehat{\odot}, [0, 0], [1, 1], \widehat{*} \rangle$ is a Kleene algebra.

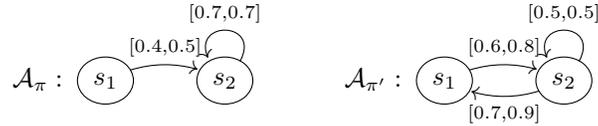
Proof. $\widehat{\max}$ trivially satisfies equations (1)-(4). According to [5, p.3224] $\widehat{\odot}$ satisfies equations (5)-(6). Equation (7) requires the result that every T-norm distributes over the maximum [15, Propostition 2.22], the rest of the proof is an exercise. Equation (8) comes from equation (7) and the commutativity of $\widehat{\odot}$. Equation (9) is also easily proved. Since for every $A \in \mathbb{U}$, $A^* = [1, 1]$, and $[1, 1]$ is the top element in \mathbb{U} , then inequation (10) is trivially satisfied. Again, since $A^* = [1, 1]$, then $A^* \widehat{\odot} x = x$ and implication (11) is satisfied. A similar argument applies to implication (12).

Since $K(\widehat{\mathbf{L}})$ is a Kleene algebra, we can canonically construct, as in (22), the space of matrices $\mathbb{M}_n(K(\widehat{\mathbf{L}}))$ (which is also a Kleene algebra).

Observation: According to Proposition 1 every operation of the Kleene algebra $K(\widehat{\mathbf{L}})$ is the best interval representation of the respective operation of $K(\mathbf{L})$. Therefore, we can say that $K(\widehat{\mathbf{L}})$ and $\mathbb{M}_n(K(\widehat{\mathbf{L}}))$ are, respectively, **the best interval representation of the Kleene algebras \mathbf{L} and $\mathbb{M}_n(K(\mathbf{L}))$** .

Notation: In order to simplify the notation we use the same symbols for the operations of Lukasiewicz Kleene algebra: $\mathbf{L} = \langle [0, 1], \max, \odot, 0, 1, * \rangle$, its interval representation $K(\widehat{\mathbf{L}}) = (\mathbb{U}, \widehat{\max}, \widehat{\odot}, 0, 1, *)$ and the corresponding spaces of matrices: $\mathbb{M}_n(K(\mathbf{L})) = (M_n(\mathbf{L}), \max, \odot, 0, 1, *)$ and $\mathbb{M}_n(K(\widehat{\mathbf{L}})) = (M_n(K(\widehat{\mathbf{L}})), \widehat{\max}, \widehat{\odot}, 0, 1, *)$.

The next automata are the interval representation of (21)



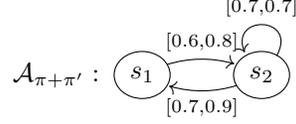
Their interval matrices are:

$$\mathcal{A}_\pi = \begin{bmatrix} (0, 0) & (0.4, 0.5) \\ (0, 0) & (0.7, 0.7) \end{bmatrix} \quad \mathcal{A}_{\pi'} = \begin{bmatrix} (0, 0) & (0.6, 0.8) \\ (0.7, 0.9) & (0.5, 0.5) \end{bmatrix}$$

The interpretation of the program $\mathcal{A}_{\pi+\pi'}$ is

$$\max \left(\begin{bmatrix} (0, 0) & (0.4, 0.5) \\ (0, 0) & (0.7, 0.7) \end{bmatrix}, \begin{bmatrix} (0, 0) & (0.6, 0.8) \\ (0.7, 0.9) & (0.5, 0.5) \end{bmatrix} \right) = \begin{bmatrix} (0, 0) & (0.6, 0.8) \\ (0.7, 0.9) & (0.7, 0.7) \end{bmatrix}$$

which represents the following weighted transition system:



4 The Price

Before we proceed, it must be clear why do we use intervals. Intervals are used in a variety of situations when it is not possible to use exact values. If the exact values can be used, then it does not make sense to use intervals.

Although it is possible to use a near exact value to represent a desired point; e.g. 3.14 would be used to represent π , the information about impreciseness is not codified by such exact value. Intervals provide such kind of information and the quality of such representation can be measured by the width of the interval: the tight is the interval the better is the representation.

Sometimes intervals are the only representation available to work with; e.g. (1) some magnetic resonance machines provide intervals for non-exact values (2) some applications in Fuzzy Systems provide intervals as inexact membership degree or as the abstraction of several membership degrees provided by different experts.

In any case, intervals are the entities provided instead of exact values. To deal with intervals a **price must be paid**; namely: not all properties of the space containing the exact values are preserved in the interval space. For example, in the case of real numbers, the respective interval representation does not satisfy the property: $x - x = 0$.

As we will see the same happens with the interval representation of the action lattice \mathbf{L} . Some properties stated in Figure 1 are satisfied by \mathbf{L} , but are not by its interval representation $\widehat{\mathbf{L}}$. Since these properties are connected with Dynamic Logics, there will be impacts of interval representation on the logical axioms. Some of these impacts are discussed below:

Observe that in the Łukasiewicz action lattice, \mathbf{L} , the equation “ $x \rightarrow x = 1$ ” is satisfied while this is not true in its interval representation: $\widehat{\mathbf{L}}$. *But this is a crucial feature of $\widehat{\mathbf{L}}$!* Take the following example: $[0.5, 0.6] \rightarrow [0.5, 0.6] = [0.6 \rightarrow 0.5, 0.5 \rightarrow 0.6] = [0.9, 1] \neq [1, 1]$. Although $1 \in [0.9, 1]$, what is happening here? Suppose that the interval $[0.5, 0.6]$ is the tightest machine interval which represents the non-finitely representable exact values in \mathbf{L} : $\frac{\pi}{6} = 0.523598775\dots$ and $\frac{\frac{\pi}{6}+0.6}{2} = 0.5617993875\dots$. Then, in order to calculate the implications: $\frac{\pi}{6} \rightarrow \frac{\pi}{6}$, $\frac{\frac{\pi}{6}+0.6}{2} \rightarrow \frac{\frac{\pi}{6}+0.6}{2}$, $\frac{\pi}{6} \rightarrow \frac{\frac{\pi}{6}+0.6}{2}$ and $\frac{\frac{\pi}{6}+0.6}{2} \rightarrow \frac{\pi}{6}$, the only way is to calculate: $[0.5, 0.6] \widehat{\rhd} [0.5, 0.6]$. In this case, $\frac{\pi}{6} \rightarrow \frac{\pi}{6} = 1$, $\frac{\frac{\pi}{6}+0.6}{2} \rightarrow \frac{\frac{\pi}{6}+0.6}{2} = 1$, $\frac{\pi}{6} \rightarrow \frac{\frac{\pi}{6}+0.6}{2} = 1$, $\frac{\frac{\pi}{6}+0.6}{2} \rightarrow \frac{\pi}{6} = 0.9617993875\dots$, and $[0.5, 0.6] \widehat{\rhd} [0.5, 0.6] = [0.6 \rightarrow 0.5, 0.5 \rightarrow 0.6] = [0.9, 1]$. Therefore, all the previous implications are

contained in the implication $[0.5, 0.6] \widehat{\rightarrow} [0.5, 0.6]$. In other words, unless an interval X has the form $[a, a]$, it does not make sense to impose $X \rightarrow X = [1, 1]$, since the same interval can be used to represent two different exact values. Therefore, the known logical laws of Dynamic Logic must be reviewed.

The price to be paid for using intervals does not stop here, in what follows we show that the structure $\widehat{\mathbb{L}} = \langle \mathbb{U}, \max, \odot, 0, 1, *, \widehat{\rightarrow}, \min \rangle$ is not an action lattice. This means that to propose a Dynamic Logic which deals with interval values some properties of action lattices must be *generalized*.

Proposition 3.

1. Equation (13) $a; x \leq b \Leftrightarrow x \leq a \rightarrow b$ (Left-residuation) fails.
2. Equation (15) $(x \rightarrow x)^* = x \rightarrow x$ fails, instead $x \rightarrow x \leq (x \rightarrow x)^*$.

Proof. 1. Make $x = [1, 1]$, then $a \odot x \leq b \Leftrightarrow x \leq a \rightarrow b$ becomes $a \leq b \Leftrightarrow a \rightarrow b = [1, 1]$ which is not true, since $x \leq x$, but, as we saw, $x \rightarrow x$ is not always equal to $[1, 1]$; make $x = [0.5, 0.6]$.
 2. Make $x = [0.5, 0.6]$. By definition $(x \rightarrow x)^* = [1, 1]$, but $x \rightarrow x = [0.9, 1]$. More generally, since $\underline{x} \rightarrow \underline{x} = \bar{x} \rightarrow \underline{x} \leq 1$ and $\overline{\underline{x} \rightarrow \underline{x}} = 1$, then $x \rightarrow x \leq [1, 1] = (x \rightarrow x)^*$.

Proposition 4. Equations (16)-(20) are satisfied.

Proof. It is well-known that the structure $\langle \mathbb{U}, \max, \min \rangle$ is a lattice.

However, it is possible to observe that some fragment of the logic generated by action lattices remains untouched and behaves as expected. For example, by considering the interpretation of the propositions $\text{Prop} = \{p, q\}$ as $V(p, s_1) = [0.1, 0.1]$, $V(q, s_1) = [0.5, 0.5]$, $V(p, s_2) = [0.7, 0.8]$ and $V(q, s_2) = [0.75, 0.75]$ we have:

$$\begin{aligned}
 (s_1 \models \langle \pi + \pi' \rangle (p \rightarrow q)) &= \max([0, 0] \odot ([0.1, 0.1] \rightarrow [0.5, 0.5]), [0.6, 0.8] \odot ([0.5, 0.5] \rightarrow [0.7, 0.8])) \\
 &= \max([0, 0], [0.6, 0.8] \odot [0.5 \rightarrow 0.7, 0.5 \rightarrow 0.8]) \\
 &= [0.6, 0.8] \odot [1, 1] \\
 &= [0.6, 0.8] \ni \frac{\sqrt{2}}{2}.
 \end{aligned}$$

5 Conclusion and Further work

Dynamic logics are very important to specify and verify properties on programs' executions. Nowadays, Dynamic logics refers to a large family of logics that have been intensively used in the verification of computational systems, that have been able to evolve and adapt to new, and complex validation challenges.

In [20, 21] there was introduced a generic method to build propositional many-valued dynamic logics, parametrized by an action lattice. Moreover, it is shown that, from beyond of these generic constructions, only the ones parametrized

by action lattices behave well, in the sense of the respect of the classic axiomatic of propositional dynamic logic [12].

We start this work by looking in a such special case, namely the Łukasiewski lattice over $[0, 1]$ (it is well known that it can be expanded to an Action lattice). This structure support a suitable framework to design and reason about systems with uncertainty degrees in transitions. However, for implementation purposes or in the verification of real systems, it is not possible to deal with exact degrees (for instance irrational numbers), being hence mandatory the use approximations. In this view we proposed in Section 3.1 an interval version of the Łukasiewski lattice. Also here, it can be defined a closure operations in order to obtain a Kleene Algebra and then the Conway's matricial constructions [9] can be applied. Although, not all works perfectly. Actually, the interval Łukasiewski Kleene algebra is not an action lattice (see Proposition 3).

This is the price to have intervals. In Section 4 we discuss some important questions related to this situation. In particular, we explain some non intuitive phenomena: for instance the implication $[a, b] \rightarrow [a, b]$ is not necessarily the top element of the lattice.

This work paves the way for an interesting research agenda. The next step will be the generalization of the intervalising process to an arbitrary Kleene algebra and to find the axiomatisation for the abstract interval pseudo-action algebras. We have already worked on this subject and some weakening of the residuum adjunction must be considered. Another important question, that we shortly consider at the end of Section 4, is how these structures constitute a sound (interval) semantics for Dynamic Logic. It can be proved that this semantics is correct in the sense that the value of a sentence in the standard case belongs to the respective interval interpretation.

Acknowledgements. R. Santiago and B. Bedregal are supported by Marie Curie project PIRSES-GA-2012-318986 GetFun funded by EU-FP7 and by the *Brazilian National Council for Scientific and Technological Development* (CNPq, Portuguese: *Conselho Nacional de Desenvolvimento Científico e Tecnológico*) under the Projects 304597/2015-5 and 307681/2012-2. This work is also financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 and by National Funds through the Portuguese funding agency, FCT - *Fundação para a Ciência e a Tecnologia* within project POCI-01-0145-FEDER-016692. A. Madeira and M. Martins are also supported by the FCT BPD individual grant SFRH/BPD/103004/2014 and UID/MAT/04106/2013 at CIDMA, respectively.

References

1. BALTAG, A., AND SMETS, S. Quantum logic as a dynamic logic. *Synthese* 179, 2 (2011), 285–306.
2. BEDREGAL, B., AND SANTIAGO, R. Some continuity notions for interval functions and representation. *Computational and Applied Mathematics* 32, 3 (2013), 435–446.

3. BEDREGAL, B. C., AND TAKAHASHI, A. Interval valued versions of t-conorms, fuzzy negations and fuzzy implications. In 2006 IEEE International Conference on Fuzzy Systems (2006), pp. 1981–1987.
4. BEDREGAL, B. R. C., AND SANTIAGO, R. H. N. Interval representations, Lukasiewicz implicators and Smets—Magrez axioms. Information Sciences 221 (2013), 192 – 200.
5. BEDREGAL, B. R. C., AND TAKAHASHI, A. The best interval representations of t-norms and automorphisms. Fuzzy Sets and Systems 157, 24 (2006), 3220 – 3230.
6. BOU, F., ESTEVA, F., GODO, L., AND RODRÍGUEZ, R. O. On the minimum many-valued modal logic over a finite residuated lattice. J. Log. Comput. 21, 5 (2011), 739–790.
7. BUSTINCE, H., BARRENECHEA, E., PAGOLA, M., FERNANDEZ, J., XU, Z., BEDREGAL, B., MONTERO, J., HAGRAS, H., HERRERA, F., AND BAETS, B. D. A historical account of types of fuzzy sets and their relationships. IEEE Transactions on Fuzzy Systems 24, 1 (Feb 2016), 179–194.
8. CIGNOLI, R., D’OTTAVIANO, I., AND MUNDICI, D. Algebraic Foundations of Many-Valued Reasoning. Trends in Logic. Springer Netherlands, 1999.
9. CONWAY, J. H. Regular Algebra and Finite Machines. Printed in GB by William Clowes & Sons Ltd, 1971.
10. FITTING, M. Many-valued modal logics. Fundam. Inform. 15, 3-4 (1991), 235–254.
11. FITTING, M. Many-valued modal logics II. Fundam. Inform. 17, 1-2 (1992), 55–73.
12. HAREL, D., KOZEN, D., AND TIURYN, J. Dynamic Logic. MIT Press, Cambridge, MA, USA, 2000.
13. HICKEY, T., JU, Q., AND VAN EMDEN, M. H. Interval arithmetic: From principles to implementation. J. ACM 48, 5 (Sept. 2001), 1038–1068.
14. HUGHES, J., ESTERLINE, A. C., AND KIMIAGHALAM, B. Means-end relations and a measure of efficacy. Journal of Logic, Language and Information 15, 1-2 (2006), 83–108.
15. KLEMENT, E. P., MESIAR, R., AND PAP, E. Triangular Norms, 1 ed. Springer, 2000.
16. KOZEN, D. On action algebras. manuscript in: Logic and Flow of Information, Amsterdam, 1991.
17. KOZEN, D. A probabilistic PDL. J. Comput. Syst. Sci. 30, 2 (1985), 162–178.
18. KOZEN, D. A completeness theorem for Kleene algebras and the algebra of regular events. Inf. Comput. 110, 2 (1994), 366–390.
19. LIAU, C. Many-valued dynamic logic for qualitative decision theory. In New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, 7th International Workshop, RSFDGrC ’99, Yamaguchi, Japan, November 9-11, 1999, Proceedings (1999), N. Zhong, A. Skowron, and S. Ohsuga, Eds., vol. 1711 of Lecture Notes in Computer Science, Springer, pp. 294–303.
20. MADEIRA, A., NEVES, R., AND MARTINS, M. A. An exercise on the generation of many-valued dynamic logics. Journal of Logical and Algebraic Methods in Programming (2016), –.
21. MADEIRA, A., NEVES, R., MARTINS, M. A., AND BARBOSA, L. S. A dynamic logic for every season. In Formal Methods: Foundations and Applications - 17th Brazilian Symposium, SBMF 2014, Maceió, AL, Brazil, September 29-October 1, 2014. Proceedings (2014), C. Braga and N. Martí-Oliet, Eds., vol. 8941 of Lecture Notes in Computer Science, Springer, pp. 130–145.
22. MOORE, R. E. Interval Arithmetic and Automatic Error Analysis in Digital Computing. Ph.D. dissertation, Department of Mathematics, Stanford Univer-

- sity, Stanford, CA, USA, Nov. 1962. Also published as Applied Mathematics and Statistics Laboratories Technical Report No. 25.
23. MOORE, R. E., AND YANG, C. T. Interval analysis I. Technical Document LMSD-285875, Lockheed Missiles and Space Division, Sunnyvale, CA, USA, 1959.
 24. MUNDICI, D. Advanced Łukasiewicz calculus and MV-algebras. Trends in Logic. Springer Netherlands, 2011.
 25. PLATZER, A. Logical Analysis of Hybrid Systems - Proving Theorems for Complex Dynamics. Springer, 2010.
 26. PRATT, V. R. Semantical considerations on floyd-hoare logic. In 17th Annual Symposium on Foundations of Computer Science, Houston, Texas, USA, 25-27 October 1976 (1976), IEEE Computer Society, pp. 109–121.
 27. PRATT, V. R. Action logic and pure induction. In JELIA (1990), vol. 478 of Lecture Notes in Computer Science, Springer, pp. 97–120.
 28. SANTIAGO, R. H. N., BEDREGAL, B. R. C., AND ACIÓLY, B. M. Formal aspects of correctness and optimality of interval computations. Formal Aspects of Computing 18, 2 (2006), 231–243.
 29. SUNAGA, T. Theory of an interval algebra and its application to numerical analysis [reprint of res. assoc. appl. geom. mem. 2 (1958), 29–46]. Japan J. Indust. Appl. Math. 26, 2-3 (10 2009), 125–143.
 30. XU, Y., RUAN, D., QIN, K., AND LIU, J. Lattice-Valued Logic: An Alternative Approach to Treat Fuzziness and Incomparability. Studies in Fuzziness and Soft Computing. Springer Berlin Heidelberg, 2012.