# Closeness and Distance Relations in Order of Magnitude Qualitative Reasoning via PDL*

A. Burrieza[1], E. Muñoz-Velasco[2], and M. Ojeda-Aciego[2]

[1] Dept. Filosofía. Universidad de Málaga. Spain
burrieza@uma.es
[2] Dept. Matemática Aplicada. Universidad de Málaga. Spain
{emilio,aciego}@ctima.uma.es

**Abstract.** The syntax, semantics and an axiom system for an extension of Propositional Dynamic Logic (PDL) for order of magnitude qualitative reasoning which formalizes the concepts of closeness and distance is introduced in this paper. In doing this, we use some of the advantages of PDL: firstly, we exploit the possibility of constructing complex relations from simpler ones for defining the concept of closeness and other programming commands such as *while ... do* and *repeat ... until*; secondly, we employ its theoretical support in order to show that the satisfiability problem is decidable and the completeness of our system. Moreover, the specific axioms of our logic have been obtained from the *minimal* set of formulas needed in our definition of qualitative sum of small, medium and large numbers. We also present some of the advantages of our approach on the basis of an example.

## 1 Introduction

The area of research within Artificial Intelligence that automates reasoning and problem solving about the physical world is called Qualitative Reasoning (QR). It creates non-numerical descriptions of systems and their behaviour, preserving important behavioural properties and qualitative distinctions. Successful application areas include autonomous spacecraft support, failure analysis and on-board diagnosis of vehicle systems, automated generation of control software for photocopiers, conceptual knowledge capture in ecology, and intelligent aids for human learning. Order of magnitude reasoning is a part of QR which stratifies values according to some notion of scale [11, 17, 19, 20].

There are different approaches in the literature [2, 12, 23] for using logic in QR that face the problem about the soundness of the reasoning supported by the formalism, and try to give some answers about the efficiency of its use. In particular, multimodal logics dealing with order of magnitude reasoning have been developed in [8, 9] defining different qualitative relations (*order of magnitude*, *negligibility*, *non-closeness*, etc.) on the basis of qualitative classes obtained by dividing the real line in intervals [22].

* Partially supported by projects TIN2006-15455-C03-01 and P6-FQM-02049.

The syntax, semantics and an axiom system for a logic which formalizes the concepts of closeness and distance are introduced in this paper. To do this, we use the advantages of Propositional Dynamic Logic [4, 14, 16], mainly the possibility of constructing complex relations from simpler ones. Some recent applications of PDL in AI can be seen in [3, 5, 6, 15]. In our case, we define the concept of closeness as a program obtained by the union of the sum of classes representing zero, positive and negative small numbers. Moreover, we introduce some nominals in order to represent the different qualitative classes, for this reason we can say that our logic is a part of Combinatory PDL [1, 18].

This work continues the line of [7] about using PDL in the framework of order of magnitude reasoning, however it introduces some differences, for example, here we use constants to represent the qualitative classes instead of the milestones which divide them, introducing an ordering only in the set of qualitative classes. This makes the approach more heavily founded on quantitativeness. Furthermore, this paper is an step forward in the formalization for two main reasons. Firstly, it gives a syntactic approach by presenting an axiom system where the specific axioms have been obtained from the *minimal* set of formulas needed in our definition of qualitative sum of small, medium and large numbers. Secondly, we have used the theoretical support of PDL in order to prove the decidability of the satisfiability problem and the completeness of this logic.

The paper is organized as follows. In Section 2, the syntax and semantics of the proposed logic is introduced, together with an example of application of our logic. In Section 3, we give an axiom system for our logic and in Section 4 the decidability of the problem of satisfiability and the completeness are proved. Finally, some conclusions and future works are discussed in Section 5.

## 2 Syntax and Semantics

In order to introduce the language of our logic, we consider a set of formulas $\Phi$ and a set of programs $\Pi$, which are defined recursively on disjoint sets $\Phi_0$ and $\Pi_0$, respectively. $\Phi_0$ is called the set of *atomic formulas* which can be thought as abstractions of properties of states. Similarly, $\Pi_0$ is called the set of *atomic programs* which are intended to represent basic instructions.

**Formulas:**

- $\Phi_0 = \mathbb{V} \cup \mathbb{C}$, where $\mathbb{V}$ is a denumerable set of propositional variables and $\mathbb{C} = \{\mathsf{nl}, \mathsf{nm}, \mathsf{ns}, \mathsf{0}, \mathsf{ps}, \mathsf{pm}, \mathsf{pl}\}$. The elements of $\mathbb{C}$ are intended to represent, respectively the qualitative classes of "negative large", "negative medium", "negative small", "zero", "positive small", "positive medium", and "positive large" numbers.
- If $\varphi$ and $\psi$ are formulas and $a$ is a program, then $\varphi \rightarrow \psi$ (propositional implication), $\bot$ (propositional falsity) and $[a]\varphi$ (program necessity) are also formulas. As usual, $\vee$ and $\wedge$ represent logical disjunction and conjunction, respectively; while $\langle a \rangle$ represents the program posibility.

**Programs:**

- $\Pi_0 = \{+_\star \mid \star \in \mathbb{C}\}$.
- If $a$ and $b$ are programs and $\varphi$ is a formula, then $(a;b)$ ("do $a$ followed by $b$"), $a \cup b$ ("do either $a$ or $b$, nondeterministically"), $a^*$ ("repeat $a$ a nondeterministically chosen finite number of times") and $\varphi?$ ("proceed if $\varphi$ is true, else fail") are also programs.

As an example of programs, we can consider $+_{\mathsf{ps}} \cup +_{\mathsf{ns}}$ and $\mathsf{ns}?; +_{\mathsf{pl}}$ in order to represent, respectively, the intuitive meanings of adding a (positive or negative) small number and adding a positive large number to a negative small number.

The possibility of construction complex programs from simpler ones allow us to define programming commands such as *while ... do* and *repeat ... until* as follows. If $\varphi$ is a formula and $a$ is a program, the program **while** $\varphi$ **do** $a$ is defined by $(\varphi?; a)^*; \neg\varphi?$ and the program **repeat** $a$ **until** $\varphi$ is given by $a; (\neg\varphi?; a)^*; \varphi?$.

We now define the *semantics* of our logic. A *model* $\mathcal{M}$ is a tuple $(W, m)$, where $W$ is a non-empty set divided in 7 qualitative classes, chosen depending on the context [21], denoted also [3] by $\{\mathsf{nl}, \mathsf{nm}, \mathsf{ns}, 0, \mathsf{ps}, \mathsf{pm}, \mathsf{pl}\}$, and $m$ is a meaning function such that $m(p) \subseteq W$, for every propositional variable, $m(\star) = \star$, for every $\star \in \mathbb{C}$ and $m(a) \subseteq W \times W$, for all program $a$. Moreover, for every formula $\varphi$ and $\psi$ and for all programs $a, b$, we have:

- $m(\varphi \rightarrow \psi) = (W \smallsetminus m(\varphi)) \cup m(\psi)$
- $m(\bot) = \varnothing$
- $m([a]\varphi) = \{w \in W : \text{for all } v \in W, \text{ if } (w,v) \in m(a) \text{ then } v \in m(\varphi)\}$
- $m(a \cup b) = m(a) \cup m(b)$
- $m(a; b) = m(a); m(b)$ (composition of relations $m(a)$ and $m(b)$)
- $m(a^*) = m(a)^*$ (reflexive and transitive closure of relation $m(a)$).
- $m(\varphi?) = \{(w, w) : w \in m(\varphi)\}$

The following properties are required for our atomic programs:

- $m(+_{\mathsf{ps}})$ is a relation on $W$ such that:

1. $m(+_{\mathsf{ps}})(\mathsf{nl}) \subseteq \mathsf{nl} \cup \mathsf{nm}$
2. $m(+_{\mathsf{ps}})(\mathsf{nm}) \subseteq \mathsf{nm} \cup \mathsf{ns}$
3. $m(+_{\mathsf{ps}})(\mathsf{ns}) \subseteq \mathsf{ns} \cup 0 \cup \mathsf{ps}$
4. $m(+_{\mathsf{ps}})(\mathsf{ps}) \subseteq \mathsf{ps} \cup \mathsf{pm}$
5. $m(+_{\mathsf{ps}})(\mathsf{pm}) \subseteq \mathsf{pm} \cup \mathsf{pl}$
6. $m(+_{\mathsf{ps}})(\mathsf{pl}) \subseteq \mathsf{pl}$

- $m(+_{\mathsf{pm}})$ is a relation on $W$ such that:

1. $m(+_{\mathsf{pm}})(\mathsf{nl}) \subseteq \mathsf{nl} \cup \mathsf{nm} \cup \mathsf{ns}$
2. $m(+_{\mathsf{pm}})(\mathsf{nm}) \subseteq \mathsf{nm} \cup \mathsf{ns} \cup 0 \cup \mathsf{ps} \cup \mathsf{pm}$
3. $m(+_{\mathsf{pm}})(\mathsf{ns}) \subseteq \mathsf{ps} \cup \mathsf{pm}$
4. $m(+_{\mathsf{pm}})(\mathsf{ps}) \subseteq \mathsf{pm} \cup \mathsf{pl}$
5. $m(+_{\mathsf{pm}})(\mathsf{pm}) \subseteq \mathsf{pm} \cup \mathsf{pl}$
6. $m(+_{\mathsf{pm}})(\mathsf{pl}) \subseteq \mathsf{pl}$

[3] By abuse of notation, we will use the same symbols to represent the qualitative classes and its corresponding formulas.

- $m(+_{\mathsf{pl}})$ is a relation on $W$ such that:

  1. $m(+_{\mathsf{pl}})(\mathsf{nm}) \subseteq \mathsf{ps} \cup \mathsf{pm} \cup \mathsf{pl}$
  2. $m(+_{\mathsf{pl}})(\mathsf{ns}) \subseteq \mathsf{pm} \cup \mathsf{pl}$
  3. $m(+_{\mathsf{pl}})(\mathsf{ps}) \subseteq \mathsf{pl}$
  4. $m(+_{\mathsf{pl}})(\mathsf{pm}) \subseteq \mathsf{pl}$
  5. $m(+_{\mathsf{pl}})(\mathsf{pl}) \subseteq \mathsf{pl}$

- $m(+_{\mathsf{ns}})$, $m(+_{\mathsf{nm}})$ and $m(+_{\mathsf{nl}})$ are given similarly and $m(+_{\mathsf{0}})$ is defined such that $m(+_{\mathsf{0}}) = \{(w, w) \mid w \in W\}$.

Notice that the properties required for the specific atomic programs are intended to reflect intuitive properties of qualitative sum. For example, $m(+_{\mathsf{ps}})(\mathsf{pl}) \subseteq \mathsf{pl}$ means that the sum of a positive small number plus a positive large number has to be a positive large number, and similarly for the rest of properties.

Given a model $\mathcal{M} = (W, m)$, a formula $\varphi$ is *true* in $u \in W$ whenever we have that $u \in m(\varphi)$. We say that $\varphi$ is satisfiable if there exists $u \in W$ such as $\varphi$ is true in $u$. Moreover, $\varphi$ is *valid in a model* $\mathcal{M} = (W, m)$ if $\varphi$ is true in all $u \in W$, that is, if $m(\varphi) = W$. Finally, $\varphi$ is *valid* if $\varphi$ is valid in all models.

The informal meaning of some of our formulas is given as follows:

- $\langle +_{\mathsf{ps}} \rangle \varphi$ is true in $u$ iff there exists $u'$, obtained by adding a positive small number to $u$, such that $\varphi$ is true in $u'$.
- $\langle \mathsf{nl}? \rangle \varphi$ is true in $u$ iff $u$ is a negative large number and $\varphi$ is true in $u$.
- $\langle +^*_{\mathsf{ps}} \rangle \varphi$ is true in $u$ iff there exists $u'$, obtained by adding a finitely many small positive numbers to $u$, such that $\varphi$ is true in $u'$.
- $[+_{\mathsf{ps}} \cup +_{\mathsf{nm}}]\varphi$ is true in $u$ iff for every $u'$, obtained by adding either a positive small number or a negative medium number to $u$, $\varphi$ is true in $u'$.

As stated above, one of the main advantages of using PDL is the possibility of constructing complex programs from basic ones. As a consequence, following the ideas presented in [7], we can use our connectives in order to represent the relations of *closeness* and *distance*. Thus, for any formula $\varphi$, we define the modal connectives $[\mathsf{c}]$ and $[\mathsf{d}]$ as follows:

$$[\mathsf{c}]\,\varphi = [+_{\mathsf{ns}} \cup +_{\mathsf{0}} \cup +_{\mathsf{ps}}]\varphi \qquad [\mathsf{d}]\,\varphi = [+_{\mathsf{nl}} \cup +_{\mathsf{pl}}]\varphi$$

The intuitive interpretation of the closeness relation is that $x$ is close to $y$ if, and only if, $y$ is obtained from $x$ by adding a small number. On the other hand, $x$ is distant from $y$ if and only if $y$ is obtained from $x$ by adding a large number.

The following example was presented in [8] for a multimodal logic. In this case, the use of PDL gives us many advantages, such as the possibility of expressing not only closeness and distance, but also the programming commands *while... do* and *repeat... until* defined above.

*Example 1.* Let us suppose that we want to specify the behaviour of a device to automatically control the temperature, for example, in a museum, subject to have some specific conditions. If we have to maintain the temperature close to some limit $T$, for practical purposes any value of the interval $[T-\epsilon, T+\epsilon]$ for small $\epsilon$ is admissible. This interval can be considered as $\mathsf{ns} \cup \mathsf{0} \cup \mathsf{ps}$ in our approach. Moreover, assume that if the temperature is out of this interval (for example, because the number of people inside the museum is changing), it is necessary to put into operation either some *heating* or *cooling* system. We also assume that, when the normal system of *cooling* or *heating* is operating, a system to maintain the humidity is needed, and when the extra system is operating, we also need an extra system of humidification. As a consequence, the qualitative classes $\mathsf{nl}, \mathsf{nm}, \mathsf{ns} \cup \mathsf{0} \cup \mathsf{ps}, \mathsf{pm}$ and $\mathsf{pl}$ can be interpreted by the formulas: VERY_COLD, COLD, OK, HOT and VERY_HOT, respectively.

We consider that program $+_{\mathsf{0}}$ means that the system is *off*; moreover $+_{\mathsf{ps}} \cup +_{\mathsf{pm}}$ and $+_{\mathsf{pl}}$, mean that a system for *heating* and *extra heating* are operating, respectively. Similarly, the programs $+_{\mathsf{nm}} \cup +_{\mathsf{ns}}$ and $+_{\mathsf{nl}}$ represent *cooling* and *extra cooling* operations, respectively.

Some consequences of the previous specification are the following:

1. $\text{HOT} \to ([+_{\mathsf{pl}}]\text{VERY\_HOT} \wedge \langle(+_{\mathsf{nm}} \cup +_{\mathsf{ns}})^*\rangle \text{OK})$
2. $[(\neg\text{OK}?; +_{\mathsf{Sys}})^*; \text{OK}?]\text{OK}$, being $+_{\mathsf{Sys}} = +_{\mathsf{nl}} \cup +_{\mathsf{nm}} \cup +_{\mathsf{ns}} \cup +_{\mathsf{ps}} \cup +_{\mathsf{pm}} \cup +_{\mathsf{pl}}$
3. $\text{VERY\_HOT} \to [(+_{\mathsf{nl}}; (\neg\text{OK}?; +_{\mathsf{nl}})^*; \text{OK}?]\text{OK}$
4. $\mathsf{0} \to [\mathsf{c}]\, \text{OK}$
5. $\text{OK} \to [\mathsf{d}]\, (\text{VERY\_COLD} \vee \text{COLD} \vee \text{HOT} \vee \text{VERY\_HOT})$

We give now the intuitive meanings for the previous formulae.

- Formula 1 means that, if the temperature is hot and the extra heating system is put into operation, then the temperature will be very hot. Moreover, if the temperature is hot, the temperature becomes OK after finitely many applications of the cooling system.
- Formula 2 says that *while* the temperature is not OK, the system has to be operating, as a consequence, we will obtain the desired temperature.
- Formula 3 is interpreted as if the temperature is very hot, *repeat* the application of the extra cooling system *until* the temperature is OK.
- Formula 4 means that every value close to the desired temperature is considered OK.
- Formula 5 can be read in this way: if the temperature is OK, for every distant value, the temperature will be either very cold or cold or hot or very hot.

Assume now that the system is more efficient (in terms of energy saving) if the temperature is close to the desired value and if the temperature is distant to these values, the system is wasting very much energy. The following formula must be true: $\text{OK} \to ([\mathsf{c}]\, \textit{efficient} \wedge [\mathsf{d}]\, \textit{warning})$, which means that for every temperature close to OK, the system is running efficiently and if the temperature is distant to OK, the system is wasting very much energy.

## 3 Axiom system

We introduce here the axiom system for our logic.

**Axiom schemata for PDL**:

**A1** All instances of tautologies of the propositional calculus.
**A2** $[a](\varphi \to \psi) \to ([a]\varphi \to [a]\psi)$
**A3** $[a](\varphi \land \psi) \leftrightarrow ([a]\varphi \land [a]\psi)$
**A4** $[a \cup b]\varphi \leftrightarrow ([a]\varphi \lor [b]\varphi)$
**A5** $[a; b]\varphi \leftrightarrow [a][b]\varphi$
**A6** $[\varphi?]\psi \leftrightarrow (\varphi \to \psi)$
**A7** $(\varphi \land [a][a^*]\varphi) \leftrightarrow [a^*]\varphi$
**A8** $(\varphi \land [a^*](\varphi \to [a]\varphi)) \to [a^*]\varphi$ (induction axiom)

**Axiom schemata for qualitative classes**:

**QE** $\mathsf{nl} \lor \mathsf{nm} \lor \mathsf{ns} \lor \mathsf{0} \lor \mathsf{ps} \lor \mathsf{pm} \lor \mathsf{pl}$
**QU** $\star \to \neg\#$ for every $\star \in \mathbb{C}$ and $\# \in \mathbb{C} - \{\star\}$

**QO1** $\mathsf{nl} \to \langle +^*_{\mathsf{ps}} \rangle\, \mathsf{nm}$
**QO2** $\mathsf{nm} \to \langle +^*_{\mathsf{ps}} \rangle\, \mathsf{ns}$
**QO3** $\mathsf{ns} \to \langle +^*_{\mathsf{ps}} \rangle\, \mathsf{0}$
**QO4** $\mathsf{0} \to \langle +^*_{\mathsf{ps}} \rangle\, \mathsf{ps}$
**QO5** $\mathsf{ps} \to \langle +^*_{\mathsf{ps}} \rangle\, \mathsf{pm}$
**QO6** $\mathsf{pm} \to \langle +^*_{\mathsf{ps}} \rangle\, \mathsf{pl}$

**Axiom schemata for specific programs**:

**PS1** $\mathsf{nl} \to [+_{\mathsf{ps}}]\, (\mathsf{nl} \lor \mathsf{nm})$
**PS2** $\mathsf{nm} \to [+_{\mathsf{ps}}]\, (\mathsf{nm} \lor \mathsf{ns})$
**PS3** $\mathsf{ns} \to [+_{\mathsf{ps}}]\, (\mathsf{ns} \lor \mathsf{0} \lor \mathsf{ps})$
**PS4** $\mathsf{ps} \to [+_{\mathsf{ps}}]\, (\mathsf{ps} \lor \mathsf{pm})$
**PS5** $\mathsf{pm} \to [+_{\mathsf{ps}}]\, (\mathsf{pm} \lor \mathsf{pl})$
**PS6** $\mathsf{pl} \to [+_{\mathsf{ps}}]\, \mathsf{pl}$

**PM1** $\mathsf{nl} \to [+_{\mathsf{pm}}]\, (\mathsf{ns} \lor \mathsf{nm} \lor \mathsf{nl})$
**PM2** $\mathsf{nm} \to [+_{\mathsf{pm}}]\, (\mathsf{nm} \lor \mathsf{ns} \lor \mathsf{0} \lor \mathsf{ps} \lor \mathsf{pm})$
**PM3** $\mathsf{ns} \to [+_{\mathsf{pm}}]\, (\mathsf{ps} \lor \mathsf{pm})$
**PM4** $\mathsf{ps} \to [+_{\mathsf{pm}}]\, (\mathsf{pm} \lor \mathsf{pl})$
**PM5** $\mathsf{pm} \to [+_{\mathsf{pm}}]\, (\mathsf{pm} \lor \mathsf{pl})$
**PM6** $\mathsf{pl} \to [+_{\mathsf{pm}}]\, \mathsf{pl}$

**PL1** $\mathsf{nm} \to [+_{\mathsf{pl}}]\, (\mathsf{ps} \lor \mathsf{pm} \lor \mathsf{pl})$
**PL2** $\mathsf{ns} \to [+_{\mathsf{pl}}]\, (\mathsf{pm} \lor \mathsf{pl})$
**PL3** $\mathsf{ps} \to [+_{\mathsf{pl}}]\, \mathsf{pl}$
**PL4** $\mathsf{pm} \to [+_{\mathsf{pl}}]\, \mathsf{pl}$
**PL5** $\mathsf{pl} \to [+_{\mathsf{pl}}]\, \mathsf{pl}$

We also consider as axioms **NS1**. . . **NS6**; **NM1**. . . **NM6** and **NL1**. . . **NL5** by changing in the previous axioms every appearance of $\mathsf{p}$ by $\mathsf{n}$ and vice versa.

**Z1** $\langle +_{\mathsf{0}} \rangle\, \varphi \to [+_{\mathsf{0}}]\, \varphi$
**Z2** $[+_{\mathsf{0}}]\, \varphi \to \varphi$

**Inference Rules**:
**(MP)** $\varphi, \varphi \to \psi \vdash \psi$ (*Modus Ponens*) **(G)** $\varphi \vdash [a]\varphi$ (generalization)

Notice that axioms **A1**. . . **A8** are classical for this type of logics. The rest ones have the following intuitive meaning:

- **QE** and **QU** mean the existence and uniqueness of the qualitative classes, respectively. **Q01–Q06** represent the ordering of these qualitative classes.

- **PS1–PS6**, **PM1–PM6** and **PL1–PL5**; the respective ones for negative numbers and **01–06** represent the desired properties of our atomic specific programs.

It is straightforward that all the previous axioms are valid formulas and that the inference rules preserve validity. For this reason, we can conclude that our system is *sound*, that is, every theorem is a valid formula.

## 4 Decidability and Completeness

In order to obtain the decidability of the satisfiability problem, we prove the *small model property.* This property says that if a formula $\varphi$ is satisfiable, then it is satisfied in a model with no more than $2^{|\varphi|}$ elements, where $|\varphi|$ is the number of symbols of $\varphi$. This result can be obtained by the technique of *filtrations* used in modal logic. However, while in modal logic it is used the concept of subformula, in PDL we have to rely on the Fisher-Lander Closure. All the results in this section can be proved in a standard way. For more details, see [14].

First of all, we define by simultaneous induction the following two functions, being $\Phi$ the set of formulas, $\Pi$ the set of programs of our logic and for every $\varphi, \psi \in \Phi$, $a, b \in \Pi$:

$$FL : \Phi \to 2^{\Phi}; \qquad FL^{\Box} : \{[a]\varphi \mid a \in \Pi, \varphi \in \Phi\} \to 2^{\Phi}$$

(a) $FL(p) = \{p\}$, for every propositional variable $p$.
(b) $FL(\star) = \star$, for all $\star \in \mathbb{C}$.
(c) $FL(\varphi \to \psi) = \{\varphi \to \psi\} \cup FL(\varphi) \cup FL(\psi)$
(d) $FL(\bot) = \{\bot\}$
(e) $FL([a]\varphi) = FL^{\Box}([a]\varphi) \cup FL(\varphi)$
(f) $FL^{\Box}([a]\varphi) = \{[a]\varphi\}$, being $a$ an atomic program.
(g) $FL^{\Box}([a \cup b]\varphi) = \{[a \cup b]\varphi\} \cup FL^{\Box}([a]\varphi) \cup FL^{\Box}([b]\varphi)$
(h) $FL^{\Box}([a; b]\varphi) = \{[a; b]\varphi\} \cup FL^{\Box}([a][b]\varphi) \cup FL^{\Box}([b]\varphi)$
(i) $FL^{\Box}([a^*]\varphi) = \{[a^*]\varphi\} \cup FL^{\Box}([a][a^*]\varphi)$
(j) $FL^{\Box}([\psi?]\varphi) = \{[\psi?]\varphi\} \cup FL(\psi)$

$FL(\varphi)$ is called the *Fisher-Lander closure* of formula $\varphi$.

The following result bounds the number of elements of $FL(\varphi)$, denoted by $|FL(\varphi)|$, in terms of $|\varphi|$. It is proved by simultaneous induction following the ideas presented in [14], taking into account our specific definition of $FL(\star) = \star$, for all $\star \in \mathbb{C}$, in the basis case of this induction.

**Lemma 1.**

*(a) For any formula $\varphi$, $|FL(\varphi)| \leq |\varphi|$.*
*(b) For any formula $[a]\varphi$, $|FL^{\Box}([a]\varphi)| \leq |a|$, being $|a|$ the number of symbols of program $a$.*

We now define the concept of filtration. First of all, given a formula $\varphi$ and a model $(W, m)$, we define the following equivalence relation on $W$:

$$u \equiv v \overset{\text{def}}{\Longleftrightarrow} \forall\, \psi \in FL(\varphi)[u \in m(\psi) \text{ iff } v \in m(\psi)]$$

The filtration structure $(\overline{W}, \overline{m})$ of $(W, m)$ by $FL(\varphi)$ is defined on the quotient set $W/\equiv$, denoted by $\overline{W}$, and the qualitative classes in $\overline{W}$ are defined, for every $\star \in \mathbb{C}$, by $\overline{\star} = \{\overline{u} \mid u \in \star\}$. Furthermore, the map $\overline{m}$ is defined as follows:

1. $\overline{m}(p) = \{\overline{u} \mid u \in m(p)\}$, for every propositional, variable $p$.
2. $\overline{m}(\star) = m(\star) = \star$, for all $\star \in \mathbb{C}$.
3. $\overline{m}(a) = \{(\overline{u}, \overline{v}) \mid \exists u' \in \overline{u}$ and $\exists v' \in \overline{v}$ such that $(u', v') \in m(a)\}$, for every atomic program $a$.

$\overline{m}$ is extended inductively to compound propositions and programs as described previously in the definition of model.

The following two Lemmas are the key of this section and are proved following also the ideas presented in [14]. To do this, we have to take into account that our definition of Fisher-Lander closure includes the qualitative classes and that the properties required in our models for atomic programs, such as $m(+_{\mathsf{ps}})(\mathsf{nl}) \subseteq \mathsf{nl} \cup \mathsf{nm}$, are maintained in the filtration structure, as a direct consequence of our previous definitions.

**Lemma 2.** *$(\overline{W}, \overline{m})$ is a finite model.*

**Lemma 3 (Filtration Lemma).** *Let $(W, m)$ be a model and $(\overline{W}, \overline{m})$ defined as previously from a formula $\varphi$. Consider $u, v \in W$.*

1. *For all $\psi \in FL(\varphi)$, $u \in m(\psi)$ iff $\overline{u} \in \overline{m}(\psi)$.*
2. *For all $[a]\psi \in FL(\varphi)$,*
   *(a) if $(u, v) \in m(a)$ then $(\overline{u}, \overline{v}) \in \overline{m}(a)$;*
   *(b) if $(\overline{u}, \overline{v}) \in \overline{m}(a)$ and $u \in m([a]\psi)$, then $v \in m(\psi)$.*

As a consequence or the previous Lemmas, we can give the following result.

**Theorem 1 (Small Model Theorem).** *Let $\varphi$ a satisfiable formula, then $\varphi$ is satisfied in a model with no more than $2^{|\varphi|}$ states.*

*Proof.* If $\varphi$ is satisfiable, then there exists a model $(W, m)$ and $u \in W$ such that $u \in m(\varphi)$. Let us consider $FL(\varphi)$ the Fisher-Lander closure of $\varphi$ and the filtration model $(\overline{W}, \overline{m})$ of $(W, m)$ by $FL(\varphi)$ defined previously. From Lemma 2, $(\overline{W}, \overline{m})$ is a finite model and by Lemma 3 (Filtration Lemma), we have that $\overline{u} \in \overline{m}(\varphi)$. As a consequence, $\varphi$ is satisfied in a finite model. Moreover, $\overline{W}$ has no more elements as the truth assignments to formulas in $FL(\varphi)$, which by Lemma 1 is at most $2^{|\varphi|}$.

In order to get the completeness of our system, we construct a *nonstandard model* from maximal consistent sets of formulas and we use a filtration lemma for nonstandard models to collapse it to a finite *standard* model. For lack of space, we present only an sketch of this proof. For more details, see [14].

A *nonstandard model* is any structure $\mathcal{N} = (N, m_\mathcal{N})$ such as it is a model in the sense of Section 2 in every respect, except that, for every program $a$, $m_\mathcal{N}(a^*)$ need not to be the reflexive and transitive closure of $m_\mathcal{N}(a)$, but only a reflexive and transitive relation which contains $m_\mathcal{N}(a)$. Given a nonstandard model $(N, m_\mathcal{N})$ and a formula $\varphi$, we can construct the filtration model $(\overline{N}, \overline{m}_\mathcal{N})$ as above, and the Filtration Lemma (Lemma 3) also holds in this case.

As said before, to obtain completeness, we define a nonstandard model $(N, m_\mathcal{N})$ as follows: $N$ contains all the maximal consistent sets of formulas of our logic and $m_\mathcal{N}$ is defined, for every formula $\varphi$ and every program $a$, by:

$$m_\mathcal{N}(\varphi) = \{u \mid \varphi \in u\}; \quad m_\mathcal{N}(a) = \{(u, v) \mid \text{for all } \varphi, \text{ if } [a]\varphi \in u \text{ then } \varphi \in v\}$$

It is easy to prove that with the previous definition, all the properties for nonstandard models are satisfied, even the ones for our specific atomic programs. Now, we can give the following completeness result.

**Theorem 2.** *For every formula $\varphi$, if $\varphi$ is a theorem then $\varphi$ is valid.*

*Proof.* We need to prove that if $\varphi$ is consistent, then it is satisfied. If $\varphi$ is consistent, it is contained in a maximal consistent set $u$, which is a state of the nonstandard model constructed above. By the Filtration Lemma for nonstandard models, $\varphi$ is satisfied in the state $\overline{u}$ of the filtration model $(\overline{N}, \overline{m}_\mathcal{N})$.

## 5 Conclusions and future work

A PDL for order of magnitude reasoning has been introduced which deals with qualitative relations as closeness and distance. An axiom system for this logic has been defined by including as axioms the formulas which express syntactically the needed properties. Moreover, we have shown the decidability of the satisfiability problem of our logic. As a future work, we are trying to extend this approach for more relations such as a linear order and negligibility, by maintaining decidability and completeness. Finally, we have planned to give a relational proof system based on dual tableaux for this logic in the line of [10, 13].

## References


1. Areces, C. and ten Cate, B. Hybrid Logics. *Handbook of Modal Logic. Studies in Logic and Practical Reasoning. Vol. 3.* Edited by Blackburn, P., Van Benthem, J. and Wolter, F. pp. 821–868. Elsevier, 2007.
2. Bennett, B., Cohn, A.G., Wolter, F. and Zakharyaschev, M. Multi-Dimensional Modal Logic as a Framework for Spatio-Temporal Reasoning. *Applied Intelligence*, 17(3): 239–251, 2002

3. Benthem, J., Eijck, J. and Kooi, B. *Logics of communication and change.* Information and Computation, 204 (11), pp. 1620-1662, 2006.
4. Blackburn, P. and Van Benthem, J. Modal Logic: A semantic perspective. *Handbook of Modal Logic. Studies in Logic and Practical Reasoning. Vol. 3.* Edited by Blackburn, P., Van Benthem, J. and Wolter, F. pp. 58–61. Elsevier, 2007.
5. Bollig, B., Kuske, D. and Meinecke, I. *Propositional dynamic logic for message-passing systems.* Lecture Notes in Computer Science 4855 pp. 303-315, 2007.
6. Bugaychenko, D., Soloviev, I. *MASL: A logic for the specification of multiagent real-time systems.* Lecture Notes in Computer Science 4696, pp. 183-192, 2007.
7. Burrieza, A., Muñoz-Velasco, E., and Ojeda-Aciego, M., *A Propositional Dynamic Logic Approach for Order of Magnitude Reasoning.*, Lecture Notes in Artificial Intelligence Vol. 5290, pp. 11-20. Springer, 2008.
8. Burrieza, A., Muñoz-Velasco, E., and Ojeda-Aciego, M., *A Logic for Order of Magnitude Reasoning with Negligibility, Non-closeness and Distance*, Lecture Notes in Artifical Intelligence 4788: 210-219, 2007.
9. Burrieza, A. and Ojeda-Aciego, M. A multimodal logic approach to order of magnitude qualitative reasoning with comparability and negligibility relations. *Fundamenta Informaticae*, 68:21–46, 2005.
10. Burrieza, A., Ojeda-Aciego, M., and Orłowska, E., *Relational approach to order of magnitude reasoning.* Lecture Notes in Computer Science 4342: 105-124, 2006.
11. Dague, P. Symbolic reasoning with relative orders of magnitude. In *Proc. 13th Intl. Joint Conference on Artificial Intelligence*, pages 1509–1515. Morgan Kaufmann, 1993.
12. Duckham, M., Lingham, J. Mason, K. andl Worboys, M. *Qualitative reasoning about consistency in geographic information.* Information Sciences, Volume 176, Issue 6, 22, pp. 601-627, 2006.
13. Golińska-Pilarek, J., and Muñoz-Velasco, E., *Relational approach for a logic for order of magnitude qualitative reasoning with negligibility, non-closeness and distance.* International Journal of Computer Mathematics. To appear, 2009.
14. Harel, D, Kozen, D. and Tiury. *Dynamic Logic* , MIT Press, 2000.
15. Heinemann, B. *A PDL-like logic of knowledge acquisition.* Lecture Notes in Computer Science 4649, pp. 146-157, 2007.
16. Mirkowska, C., Salwicki, A. *Algorithmic Logic* Kluwer Academic Publishers Norwell, 1987.
17. Nayak, P. *Causal Approximations.* Artificial Intelligence 70. 277-334, 1994.
18. Passy, S. and Tinchev, T. An essay in combinatory dynamic logic. *Information and Computation* Volume 93 , Issue 2 (August 1991), pp 263 - 332, 1991.
19. Raiman, O. Order of magnitude reasoning *Artificial Intelligence.* 51: 11-38, 1991.
20. Sanchez, M. Prats, F. and Piera, N. Una formalización de relaciones de comparabilidad en modelos cualitativos *Boletín de la AEPIA (Bulletin of the Spanish Association for AI)* 6: 15-22, 1996.
21. Travé-Massuyès, L., Ironi, L. and Dague P., *Mathematical Foundations of Qualitative Reasoning.*, AI Magazine, American Asociation for Artificial Intelligence, 91–106, 2003.
22. Travé-Massuyès, L., Prats, F., Sánchez, M. and Agell, N. Relative and absolute order-of-magnitude models unified. In *Annals of Mathematics and Artificial Intelligence* 45: 323-341, 2005.
23. Wolter, F. and Zakharyaschev, M. Qualitative spatio-temporal representation and reasoning: a computational perspective. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium.* Morgan Kaufmann, 2002.