Formalizing Distinguishability: A Contexts Based Case

Raúl Ramos[1,2] and **Álvaro Barreiro**[2]
[1] CERN, European Laboratory for Particle Physics, Geneva, Switzerland
[2] AI Lab, Department of Computer Science, University of A Coruña, Spain
email: `Raul.Ramos@cern.ch`, `barreiro@udc.es`

Abstract

We aim at formalizing the notion of distinguishability. Intuitively, *distinguishability* refers to the transitions between different levels of detail with which different agents refer to the same objects. We frame our work within the notion of *contexts*, as a way to provide agents with local reasoning capabilities and to express their compatibilities. We believe distinguishability plays an important role in any effort to build or describe interaction between arbitrary agents. We show as well how the notion of distinguishability can be used by agents to state in a local way their beliefs about other agents without having to reify their formulae. Finally, after giving a generic formalization and a simple example, we show how distinguishability can be formalized within a certain *logic of context*.

Keywords: contexts, logics of context, distinguishability, granularity.

1 Introduction

We firstly aim at formalizing the notion of distinguishability. We believe this notion may be very useful when modeling interactions between arbitrary agents. Intuitively, *distinguishability* refers to the transitions between the levels of detail with which different agents refer to the same objects. Usually, one may go from a finer level of detail to a coarser one by rendering indistinguishable certain objects which are distinguishable in the finer level. Higher and lower levels of details are said to have finer and coarser granularities respectively. However, we are not assuming any absolute hierarchy of granularities and, for instance, a set of objects may be distinguishable for an agent but not for another one while a different set of objects may be indistinguishable for the first agent but not for the second one. Also, different agents may represent different physical entities, levels of abstraction, perspectives of the same world, etc. An example in [Hobbs, 1985] is quite illustrative: *"When we are planning a trip it is sufficient to think of a road as a one dimensional curve; when we are crossing the road we think of it as a surface and when we are digging a hole it becomes a volume for us. When we are driving down a road we alternate among these granularities, sometimes conscious of our progress along the one dimensional curve, sometimes making adjustments in our position in the surface, sometimes slowing down for bumps."* In this case, sometimes three dimensional information is rendered indistinguishable in two dimensional representations and other times two dimensional information gives us valuable hints about the underlying objects which are only distinguishable in the three dimensional world.

In general, we believe the notion of distinguishability plays a very important role in any effort to build or describe interactions between arbitrary agents in the sense stated in [Jennings, 1999]: *"agents need the computational apparatus to make context-dependent decisions about the nature and scope of their interactions and to initiate (and respond to) interactions that were not foreseen at design time."*

Our second aim is to show how an adequate formalization of the notion of *distinguishability* could provide the foundations for arbitrary *local agents* to use the mechanisms of *logics of context* to state hypothesis about their relation with other (local) agents.

We take the approach of *contexts* as founded in [McCarthy, 1987] and [Giunchiglia and Ghidini, 1998] and taken into practice in the *logics of contexts* (LCs) ([Buvac, 1996], [Ghidini and Serafini, 1998a] or [Nayak, 1995]). We consider agents formalized as contexts. In the field of agents, contexts have been used to formalize reasoning about belief ([Giunchiglia and Serafini, 1994] and [Benerecetti *et al.*, 1997]) or information integration ([Ghidini and Serafini, 1998b]). Different contexts typically refer to different agents that are somehow related by sharing different views of the same world, holding complementary or partial information, etc. LCs provide mechanisms for (1) stating formulae in different contexts, (2) reasoning within a certain context without disturbing other contexts and (3) establishing relations among contexts. From a model theoretic point of view the underlying idea of any LC is that of specifying models which are satisfied simultaneously in various contexts. This idea is rooted in the fact that, whenever there is something common (shared) between agents, their conceptualizations are necessarily related, in the sense that knowing something about an agent may help us know something else about some other agent. This idea is also set forth in [Giunchiglia and Ghidini, 1998]. Usually, in order to state relations between agents one has to be *above* them, or otherwise agents have to be *contextually omniscient* (know everything about each other). In many occasions neither of the cases is realistic nor possible. In this sense an agent having access only to its own knowings is said to be *strictly local*.

The general scenario that we are considering is the following. Assume a set of agents living in the same world, holding information from different viewpoints or having something in common which makes them related to each other somehow. Formally, each agent uses at least one context to represent and manipulate its knowledge. Different contexts use different first order languages and the language(s) of and agent are the language(s) of the context(s) it uses. An agent can access *only* its own contexts (they are *local* contexts) although every agent is aware that there are other agents living in the same world. Nobody has complete knowledge. We put ourselves in the shoes of a certain agent $\mathcal{A}$. Agent $\mathcal{A}$ would like to write formal expressions using some LC about its relation with some other certain agent $\mathcal{B}$. $\mathcal{A}$ knows that $\mathcal{B}$ lives in the same world and, eventually, will interact with it, so it would like to formalize that interaction from its own point of view. Expressions in any LC relating models of two contexts require writing formulae in the local language of *both* contexts and then putting them together in an LC expression. If $\mathcal{A}$ was contextually omniscient (had access to the language of any other context, including those of agent $\mathcal{B}$) it could write LC expressions straight away stating its beliefs about $\mathcal{B}$. Since this is not the case, $\mathcal{B}$'s language is not accessible and $\mathcal{A}$ cannot write LC expressions about $\mathcal{B}$. In this situation $\mathcal{A}$ can only try to *guess* what $\mathcal{B}$'s language would be like. *Then* $\mathcal{A}$ could write LC expressions about $\mathcal{B}$ by creating a new language combining its own with these guessings or hypothesis. These LC expressions about $\mathcal{B}$ represent $\mathcal{A}$'s own subjective view of its relation with $\mathcal{B}$. *Guessings* or *hypothesis* are made by $\mathcal{A}$ based on its own intuition, partial information given by an external source, empirical observation [Jong, 1999], or whatever. Since both agents may eventually interact, $\mathcal{A}$ may want to verify or refine its hypothesis by interacting with $\mathcal{B}$ in an appropriate way.

Summarizing, $\mathcal{A}$ is facing a three step process: (1) making hypothesis about $\mathcal{B}$'s language, (2) writing down LC expressions about $\mathcal{B}$ using $\mathcal{A}$'s own language and $\mathcal{B}$'s guessed language and (3) verifying or refining the hypothesis through interaction with $\mathcal{B}$. We aim at providing a framework where to perform steps (1) and (2) based on the notion of distinguishability. Next section provides an initial formalization of the notion of distinguishability and applies it to a sample case. In section 3 we deal with distinguishability from the point of view of logics of context. Section 4 poses various implementational issues and section 5 discusses future implications of our work.

2 Formalizing distinguishability

In order to formalize distinguishability we use the notion of grouping constants based on relevant predicates as firstly proposed in [Hobbs, 1985]. Let A denote a context used by agent $\mathcal{A}$, and $L_\mathsf{A} = (\mathbb{C}_\mathsf{A}, \mathbb{P}_\mathsf{A}, \mathbb{F}_\mathsf{A})$ the first order language of A, with $\mathbb{C}_\mathsf{A}$ the set of constants of L_A, $\mathbb{P}_\mathsf{A}$ the set of predicates and $\mathbb{F}_\mathsf{A}$ the set of functions. $\mathcal{M}(L_\mathsf{A})$ denotes a set of models over L_A, and $\Theta(\mathcal{M}(L_\mathsf{A}))$ denotes the set of sentences of L_A satisfied in $\mathcal{M}(L_\mathsf{A})$.

2.1 Indistinguishability Relations

Let $C = \{C_1, C_2, ..., C_n\}$ be a partition of $\mathbb{C}_\mathsf{A}$. We say that $a, b \in \mathbb{C}_\mathsf{A}$ are **indistinguishable** with respect to C, in symbols $a \overset{C}{\sim} b$, iff they both belong to the same C_i. We say that $\overset{C}{\sim}$ is an **indistinguishability relation**. Also, $\overset{C}{\sim}$ is just the equivalence relation whose equivalence classes are the elements of C and we denote by $\kappa(a)$ the equivalence class to which a belongs under $\overset{C}{\sim}$.

At a given moment, C can be defined in convenience with the specific case we may be dealing with. It could be defined by explicitly enumerating every C_i and its elements or also intensionally. For instance, by choosing a certain set of *relevant predicates* whose behavior with constants determines to which C_i each constant belongs (see [Hobbs, 1985]).

The idea is that if $\mathcal{A}$ wants to make hypothesis about its relation to another agent $\mathcal{B}$ in the sense described in the previous section, $\mathcal{A}$ will create a new context Bh containing $\mathcal{B}$'s guessed language. The language of Bh is $L_\mathsf{Bh} = (C, \mathbb{P}_\mathsf{Bh}, \mathbb{F}_\mathsf{Bh})$ with $\mathbb{P}_\mathsf{Bh}$ and $\mathbb{F}_\mathsf{Bh}$ containing the predicates and functions that $\mathcal{A}$ suspects $\mathcal{B}$ has. Note that the constants of this language are the equivalence classes from $\overset{C}{\sim}$ themselves. Then, $\mathcal{A}$ needs a way to build expressions defining the predicates and functions included in Bh so that they are related to the knowledge contained in A. Since these expressions need to combine terms and predicates from L_A and L_Bh we define a new language where to state them: $L_\mathsf{ABh} = (\mathbb{C}_\mathsf{A} \cup C, \mathbb{P}_\mathsf{A} \cup \mathbb{P}_\mathsf{Bh}, \mathbb{F}_\mathsf{A} \cup \mathbb{F}_\mathsf{Bh} \cup \{\kappa\})$

This way, $\mathcal{A}$ can now use L_ABh to state its beliefs about $\mathcal{B}$ without having to reify $\mathcal{B}$'s formulae but just building first order formulae in L_ABh. The following example illustrates these ideas.

2.2 Example

Assume $\mathcal{A}$ is conceptualizing the world of figure 1, encapsulating this knowledge in context A. The language L_A of A contains $\mathbb{C}_\mathsf{A} = \{0..19\}$, $\mathbb{P}_\mathsf{A} = \{occupied\}$ and $\mathbb{F}_\mathsf{A} = \varnothing$, with the following interpretation: the elements of $\mathbb{C}_\mathsf{A}$ stand for the cells as numbered in figure 1 and $occupied(x) \in \Theta(\mathcal{M}(L_\mathsf{A}))$ if the cell corresponding to x in figure 1 is black.

$\mathcal{A}$ knows that there is another agent, $\mathcal{B}$, conceptualizing the same world and suspects that $\mathcal{B}$ is doing

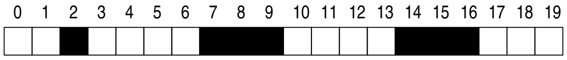

Figure 1: World conceptualized by $\mathcal{A}$

it with a granularity twice as coarser with some degree of shading. Graphically, what $\mathcal{A}$ thinks that $\mathcal{B}$ is conceptualizing out of figure 1 looks like this:

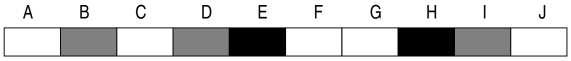

Figure 2: What $\mathcal{A}$ suspects that $\mathcal{B}$ is conceptualizing

so that cells in figure 1 are grouped two by two. The resulting cell in figure 2 is colored gray if only one of the corresponding cells of figure 1 is black, it is colored black if both of them are black, and it is left white otherwise.

Although $\mathcal{A}$ does not know anything about $\mathcal{B}$'s language, it would like to formalize its suspicions about it. In particular, $\mathcal{A}$ suspects that $\mathcal{B}$'s language should have constants representing the grid with a coarser granularity, and predicates corresponding to the notions of occupation and shading. Of course, this implies that models on the languages of both $\mathcal{A}$ and $\mathcal{B}$ are related, i.e, if $occupied(2)$ and $\neg occupied(3)$ are satisfied in some model on L_{A}, then there must be some model on $\mathcal{B}$'s language satisfying predicates which must look like $occupied(B)$ and $gray(B)$.

Once these suspicions have been formalized, $\mathcal{A}$ could try to verify or refine them by interacting with $\mathcal{B}$ in the appropriate way. But first, $\mathcal{A}$ has to be able to **state** its suspicions formally. To do this, $\mathcal{A}$ builds C in the following way:

$$C = \{\{0,1\},\{2,3\},\{4,5\},...\{18,19\}\} \tag{1}$$

naming the elements of C with alphabet letters, so that $A = \{0,1\}$, $B = \{2,3\}$, etc. C renders the elements of $\mathbb{C}_{\mathsf{A}}$ indistinguishable two by two. Note that we have defined C by explicitly enumerating its C_i and their elements. The set $\{A, B, ...\}$ represents the constants $\mathcal{A}$ suspects $\mathcal{B}$ has. Next, $\mathcal{A}$ builds a new context, Bh, whose language is $L_{\mathsf{Bh}} = (C, \mathbb{P}_{\mathsf{Bh}}, \mathbb{F}_{\mathsf{Bh}})$, with $\mathbb{P}_{\mathsf{Bh}} = \{b\_occupied, b\_gray, b\_black\}$ and $\mathbb{F}_{\mathsf{Bh}} = \varnothing$. Finally, $\mathcal{A}$ uses L_{ABh} to define the semantics of the new predicates to capture what $\mathcal{A}$ suspects about $\mathcal{B}$:

$$b\_occupied(x) \Leftrightarrow \exists z \;\; \kappa(z) = x \wedge occupied(z) \tag{2}$$
$$b\_gray(x) \Leftrightarrow \exists! z \;\; \kappa(z) = x \wedge occupied(z) \tag{3}$$
$$b\_black(x) \Leftrightarrow \forall z \;\; \kappa(z) = x \Rightarrow occupied(z) \tag{4}$$

With this, $\mathcal{A}$ has concrete expressions relating its beliefs about $\mathcal{B}$ to its own language and models.

Alternatively, we could also empower context A with a function, $\mathbb{F}_{\mathsf{A}} = \{div2\}$, where $div2(x)$ associates x with the integer part of $x/2$. This way $\mathcal{A}$ could have defined C intensionally: $\forall a, b \in \mathbb{C}_{\mathsf{A}}, a \overset{C}{\sim} b$ iff $div2(a) = div2(b)$, and even define a new function $b\_div2 \in \mathbb{F}_{\mathsf{Bh}}$ so that it conveys the same meaning as $div2$ but referred to the new constants, using the following expression of L_{ABh}:

$$b\_div2(\kappa(x)) = \kappa(div2(x)) \tag{5}$$

for instance, $b\_div2(E) = \kappa(div2(8)) = \kappa(div2(9)) = \kappa(4) = C$, which corresponds to the notion of integer division applied to the new alphabetic constants.

3 Distinguishability within logics of contexts

In previous section we have shown how to formalize a certain notion of distinguishability within a general framework. We used a generic notion of context to isolate the knowledge of an agent from the hypothesis it makes about the other agent. Since logics of contexts also provide mechanisms to state arbitrary relations between contexts, we would like to frame the formalization of previous section within some *logic of context*. In a general sense, any formalization of distinguishability should reduce to a certain class of expressions built within a logic of context. But before, we need to introduce some notions of contexts in sections 3.1 and 3.2.

3.1 Local Model Semantics

We take the model-theoretic framework set forth in Local Model Semantics (LMS,[Giunchiglia and Ghidini, 1998]). LMS provides concrete semantics which captures the two intuitions underlying contextual reasoning: (i) reasoning and representation is mainly *local* and (ii) there is compatibility among reasoning and representations happening in different contexts. LMS assumes each context is using a first order language to describe what is true within in. Then, given two contexts, it formalizes these two intuitions by defining a set of abstract **Compatibility Constraints** between the local models of both contexts. These constraints are represented by a **Compatibility Relation C** so that **C** will contain a set of pairs, each pair containing a set of models of one context and a set of models of the other context. Intuitively, one can think of each pair of **C** as containing models which occur *simultaneously* in both contexts.

3.2 Distributed First Order Logics

We will use DFOL [Ghidini and Serafini, 1998a] as the logic of context within which we formalize distinguishability. DFOL takes into practice the ideas of LMS by defining concrete mechanisms to state compatibility constraints. Briefly, we summarize DFOL. Let $\{L_i\}$ be a family of first order languages defined over a set of indexes $\mathcal{I}$. Each language L_i partially describes the world from a certain perspective (agent i's point of view). Each L_i is independent from each other, so the same formula ϕ may occur in different languages, possibly representing different facts. To differentiate between formulae of different languages, DFOL labels any formula with the index of the language where it occurs. So $\langle \phi, i \rangle$ denotes the formula ϕ and the fact that ϕ is a formula of language L_i. The structure on which the languages $\{L_i\}$ are interpreted is made of two constituents: (i) the local interpretation for each L_i; and (ii) for each pair $\langle L_i, L_j\rangle$ a binary relation between the elements of their interpretation domains.

DFOL models

First, let M_i be the set of first order models of L_i on a given domain $\mathbf{dom}_i$. Any $m \in M_i$ is a **local model** (of L_i). The usual definition of a first order model applies: m is a pair $\langle \mathbf{dom}_i, \mathcal{I}\rangle$, where dom_i is the domain of interpretation (or universe) of L_i and $\mathcal{I}$ is the interpretation function.

Second, there is a family of relations between objects of different interpretation domains. Each of these relations is a **domain relation**. Formally, a domain relation r_{ij} from $\mathbf{dom}_i$ to $\mathbf{dom}_j$ is a subset of $\mathbf{dom}_i \times \mathbf{dom}_i$. A pair $\langle d, d'\rangle \in r_{ij}$ means that, from the point of view of j, d in $\mathbf{dom}_i$ is somehow related to d' in $\mathbf{dom}_j$ (maybe they refer to the same object of some common world, or they denote different parts or views of the same object, etc.)

Finally, a DFOL model $\mathcal{M}$ (for $\{L_i\}$) is a pair $\langle\{S_i\}, \{r_{ij}\}\rangle$ where every $S_i \subseteq M_i$ is a set of local models of L_i with the same domain of interpretation $\mathbf{dom}_i$ and every r_{ij} is a domain relation from i to j.

DFOL Constraints

Once this is set up, DFOL establishes the mechanisms to enable formal reasoning between different first other theories (the knowledge of each agent) by providing a way to specify constraints in DFOL models. Intuitively, these constraints describe the sets of local models that hold *simultaneously* in various contexts. DFOL defines two kinds of constraints: domain constraints and view constraints.

A **domain constraint** from L_i to L_j is an expression of the form T or S. Intuitively, T from i to j captures the fact that, from j's perspective, the set of objects in $\mathbf{dom}_i$ is contained in $\mathbf{dom}_j$, and conversely for S.

A **view constraint** from L_i to L_j is an expression on the form

$$\langle \phi(x_1, ..., x_n), i\rangle \rightarrow \langle \psi(x_1, ..., x_n), j\rangle$$

where $\phi(x_1, ..., x_n)$ and $\psi(x_1, ..., x_n)$ are formulae of L_i and L_j, respectively. Intuitively, the view constraint $\langle \phi(x_1, ..., x_n), i\rangle \rightarrow \langle \psi(x_1, ..., x_n), j\rangle$ captures the fact that, from the point of view of j, the set of tuples of objects of $\mathbf{dom}_i$ which satisfy $\phi(x_1, ..., x_n)$ in L_i corresponds (are mapped by j via r_{ij} in) to a set of tuples which satisfy $\psi(x_1, ..., x_n)$ in L_j.

Then, satisfiability of constraints is based on whatever domain relation r_{ij} taken into consideration at a given moment:

Satisfiability of a Domain Constraint: The domain relation r_{ij} satisfies the domain constraint T iff for any $d \in \mathbf{dom}_i$ there is a $d' \in \mathbf{dom}_j$ such that $\langle d, d'\rangle \in r_{ij}$. And conversely for S.

Satisfiability of a View Constraint: Let S_i and S_j be two sets of first order models of L_i and L_j, respectively, and r_{ij} be a domain relation. The tuple $\langle S_i, S_j, r_{ij}\rangle$ satisfies the view constraint $\langle \phi(x_1, ..., x_n), i\rangle \rightarrow \langle \psi(x_1, ..., x_n), j\rangle$ iff for any $\langle d_k, d'_k\rangle \in r_{ij}$ $(1 \le k \le n)$, $S_i \vDash \phi(d_1, ..., d_k)$ implies that $S_j \vDash \psi(d'_1, ..., d'_k)$.

3.3 Distinguishability in DFOL

In terms of what was formalized in section 2, for any $\overset{C}{\sim}$ we can build a correspondient domain relation, denoted by r_C, in the following way:

$$r_C = \{\langle c, \kappa(c)\rangle | c \in \mathbb{C}_{\mathsf{A}}\} \tag{6}$$

which satisfies domain constraint T. For instance, the domain relation following the indistinguishability relation built from C as defined in expression (1) contains the following pairs:

$$r_C = \{\langle 0, A\rangle, \langle 1, A\rangle, \langle 2, B\rangle, \langle 3, B\rangle, ..., \langle 19, J\rangle\} \tag{7}$$

In the example of previous section, $\mathcal{A}$ is trying to establish its relation with $\mathcal{B}$ by making hypothesis about $\mathcal{B}$'s beliefs. As in previous section, A denotes the context where $\mathcal{A}$ represents its knowledge, and Bh denotes $\mathcal{A}$'s hypothesis about $\mathcal{B}$. Similarly, r_C can also be denoted by r_{ABh}. Once we have the domain relation we can build view constraints to capture the intended meaning of expressions (2) through (4) so that we have everything under DFOL. Expression (2) corresponds to the following view constraint:

$$\langle \mathit{occupied}(x), \mathsf{A}\rangle \rightarrow \langle \mathit{b\_occupied}(x), \mathsf{Bh}\rangle \tag{8}$$

We would like to do the same for expressions (3) and (4), however DFOL provides a fixed definition on how the domain relation participates in the satisfiability of a view constraint. The difference among expressions (2) through (4) is that they quantify differently over the elements of r_{ABh} in which the constants of $\mathcal{A}$ that satisfy $\langle \mathit{occupied}(x), \mathcal{A}\rangle$ participate. In the semantics defined by DFOL for view constraints, any view constraint quantify over those elements in the same way. Therefore, we have to find some other way to state expressions (3) and (4) in DFOL. For instance, assuming $\mathcal{A}$ also has the functions $+$ and $*$ defined with the usual meaning, expression (4) can be stated like this:

$$\langle \mathit{occupied}(\mathit{div}2(x) * 2) \wedge \mathit{occupied}(\mathit{div}2(x) * 2 + 1), \mathsf{A}\rangle \rightarrow \langle \mathit{b\_black}(x), \mathsf{Bh}\rangle \tag{9}$$

The semantics of expression (4) and the fixed definition of the way the domain relation participates in the semantics of view constraints makes us build expression (9) like this. This way, view constraints (8) and (9) are satisfied by the domain relation r_{ABh} of expression (7) and the respective models on A and Bh.

With this, our goal was to show how the idea of *distinguishability* can enable us to use a logic of context *from within* an agent. In the case of DFOL, we can create meaningful view and domain constraints and domain relations from within an agent without having to assume any contextually omniscient entity. In a certain sense, what we have done is stating in a local way what previously had to be stated at a contextually omniscient level.

4 Implementational Issues

We believe the notion of distinguishability as presented in this paper can be successfully applied to build or describe interactions in heterogeneous multi-agent scenarios where no common ontology or communication semantics can be assumed. For this cases, we aim at obtaining an interaction protocol, with distinguishability at its basis, that would enable highly heterogeneous agents to reach effective communication.

To illustrate this idea let's consider a distributed task scheduling scenario where a set of agents must agree on a schedule to work together towards performing some certain global task. There are several mechanisms by which this can be achieved [Parunak, 1998] such as auction-based mechanisms [Walsh *et al.*, 1998] where agents advertise (1) what they need in order to perform their individual task and (2) the result of this task once it is carried out. After some negotiation rounds the outcome is a global organization of the agents so that the overall task can be achieved. However, if agents are too heterogeneous this may be problematic. Consider the case of agents $\mathcal{A}$ and $\mathcal{B}$ of section 2 working together to obtain a certain cell configuration. After the negotiation round they agree that *$\mathcal{A}$ turns on the second cell from the right, and $\mathcal{B}$ turns on the second cell from the left.* However, this statement has a different meaning for $\mathcal{A}$ or $\mathcal{B}$ if they are not mutually aware of their different granularities. This suggests that in cases such as this one, the negotiation round should be preceded by some certain interaction aimed at establishing the concepts of each agent that are distinguishable for the other agents and the ones that are not.

Certain insights behind the ideas presented here are similar to those found in previous works concerning the notion of *abstraction*. Those works deal in general with mappings between systems and respond to several motivations such as reducing search spaces of problems [Giunchiglia and Walsh, 1992] or building models for approximate reasoning [Imielinski, 1987]. Also, some approaches deal with abstraction at a semantical (model based) level [Nayak and Levy, 1995] while others do it at a syntactical level. See [Giunchiglia *et al.*, 1997] for a more complete survey. The motivation of our work is different in the sense that we are looking for ways for heterogenous agents to share knowledge assuming there is no *preagreed* ontologies. However, some of the mechanisms developed in different approaches of abstraction provide valuable insights into certain aspects of our work, such as automatic ways to generate expressions in L_{ABh} under certain conditions or translate theories of A into theories of Bh. These aspects may play a crucial role in the implementation of the notions presented in this paper.

5 Discussion and future work

The purpose behind section 3.3 was to show how the notion of distinguishability could be formalized within a certain logic of context and we used DFOL for that. Besides different technical issues we might have encountered, we could have used other logic, such as the Quantificational Logic of Context (QLC, [Buvac, 1996]) to fulfill the same background goals. QLC and DFOL offer different approaches to provide specific means for stating relations between contexts. Although they start off from different underlying assumptions, both of them provide concrete logics which follow the principles behind the notion of *context* and, in this sense, they come to fill a similar gap. This common point constitutes an appropriate framework where to start dealing with the notion of distinguishability as presented in this paper. In the way, we have found that DFOL's fixed definition regarding the participation of domain relations in the semantics of view constraints allows us to express in a direct way notions such as *occupied* as shown above.

The example in section 2.2 is simple enough to focus the attention on an elementary formalization of the notion of distinguishability. However, we believe distinguishability is general enough to be present in many situations where there is some kind of compatibility between participating entities. Future work heads towards finding significant applications within the field of agents where the notion of distinguishability can be shown to be essential in their implementation, in the line of what was suggested in section 4. In some sense, with that we also intend to find other important fields where distinguishability may play a significative role.

6 Conclusions

In this paper we have formalized the intuitive notion of distinguishability applied to the constants of a first order language. It has been shown how this formalization can be used as the basis for a mechanism to allow local agents (contexts) to make hypothesis about other interacting agents. Beliefs about other agents can be now stated locally as regular first order formulae without having to reify their formulae. Since the notion of distinguishability between agents falls within the principles of the logics of contexts we have also formalized it within a certain logic of context (DFOL) and, incidentally, we have seen how certain simple notions can be stated more directly in DFOL than others.

Acknowledgements

This work is partially supported by projects PB97-0228 from *Ministerio de Educacion y Cultura*, Spain and PGIDT99PXI10201B from *Xunta de Galicia*, Spain.

References

[Benerecetti *et al.*, 1997] M. Benerecetti, A. Cimatti, E. Giunchiglia, F. Giunchiglia, and L. Serafini. Formal specification of beliefs in multi-agent systems. In *Intelligent Agents III - Proc. 3rd International Workshop on Agent Theories, Architectures, and Languages (ATAL-96). Lecture Notes in Artificial Intelligence*, volume 1193, pages 117–130, 1997.

[Buvac, 1996] S. Buvac. Quantificational logic of context. In *Proc. 13^{th} National Conference on Artificial Intelligence (AAAI-96)*, pages 600–606, Portland, Oregon, 1996.

[Ghidini and Serafini, 1998a] C. Ghidini and L. Serafini. Distributed first order logics. In *Proc. Frontiers of Combining Systems (FroCoS'98)*, Amsterdam, The Netherlands, 1998.

[Ghidini and Serafini, 1998b] C. Ghidini and L. Serafini. Information integration for electronic commerce. In *Proc. Workshop on Agent Mediated Electronic Trading (AMET'98). Lecture Notes in Artificial Intelligence*, volume 1571, Minneapolis, Minnesota, 1998.

[Giunchiglia and Ghidini, 1998] F. Giunchiglia and C. Ghidini. Local Models Semantics, or Contextual Reasoning Locality + Compatibility. In *Proc. 6^{th} International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 282–289, Trento, Italy, 1998.

[Giunchiglia and Serafini, 1994] F. Giunchiglia and L. Serafini. Multilanguage hierarchical logics (or: how we can do without modal logics). *Artificial Intelligence*, 65(1):29–70, 1994.

[Giunchiglia and Walsh, 1992] Fausto Giunchiglia and Toby Walsh. A theory of abstraction. *Artificial Intelligence*, 57(2-3):323–389, 1992.

[Giunchiglia *et al.*, 1997] Fausto Giunchiglia, Adolfo Villafiorita, and Toby Walsh. Theories of abstraction. *AI-Communications*, 10:167–176, 1997.

[Hobbs, 1985] J.R. Hobbs. Granularity. In A. Joshi, editor, *Proc. 9^{th} International Joint Conference on Artificial Inteligence (IJCAI-85)*, pages 432–435, Los Angeles, California, 1985.

[Imielinski, 1987] Tomasz Imielinski. Domain abstraction and limited reasoning. In *Proc. 10^{th} International Joinf Conference on Artificial Intelligence*, pages 997–1003, Milan, Italy, 1987.

[Jennings, 1999] N.R. Jennings. Agent-based computing: Promise and perils. In *Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 1429–1436, Stockholm, Sweeden, 1999.

[Jong, 1999] E.D. De Jong. Autonomous concept formation. In T. Dean, editor, *Proc. 16^{th} International Joint Conference on Artificial Intelligence, IJCAI'99*, pages 344–349, San Francisco, CA, 1999.

[McCarthy, 1987] J. McCarthy. Generality in artificial intelligence. *Communications of ACM*, 30(12):1030–1035, 1987.

[Nayak and Levy, 1995] P. Pandurang Nayak and Alon Y. Levy. A sematic theory of abstraction. In Chris S. Mellish, editor, *Proc. 14^{th} International Joint Conference on Artificial Intelligence*, pages 196–203, San Mateo, August 20-25 1995.

[Nayak, 1995] P. Pandurang Nayak. Representing multiple theories. In *Proc. of AAAI-94*, pages 1154–1160, Seattle, WA, 1995.

[Parunak, 1998] H.V.D. Parunak. Characterizing multi-agent negotiation, Working Group Summary. In *International Workshop on Multi-Agent Systems, IWMAS-98*, 1998. http://www.erim.org/~van/IWMAS98_Characterize.pdf .

[Walsh *et al.*, 1998] William Walsh, Michael Wellman, Peter Wurman, and Jeffrey MacKie-Mason. Some economics of market-based distributed scheduling. In *Proc. 18^{th} International Conference on Distributed Computing Systems*, pages 612–621, Amsterdam, The Netherlands, 1998.