Elaborating Analogies from Conceptual Models

George Spanoudakis

Department of Computer Science, The City University, U.K
Email: gespan@cs.city.ac.uk

Panos Constantopoulos

Department of Computer Science, University of Crete, Greece and
Institute of Computer Science, Foundation for Research and Technology-Hellas
Email: panos@ics.forth.gr

Abstract. This paper defines and analyses a computational model of similarity which detects analogies between objects based on conceptual descriptions of them, constructed from *classification*, *generalization* relations and *attributes*. Analogies are detected(elaborated) by functions which measure conceptual distances between objects with respect to these semantic modelling abstractions. The model is domain independent and operational upon objects described in non uniform ways. It doesn't require any special forms of knowledge for identifying analogies and distinguishes the importance of distinct object elements. Also, it has a polynomial complexity. Due to these characteristics, it may be used in complex tasks involving intra or inter-domain analogical reasoning. So far the similarity model has been applied in the domain of software engineering. First, to support the specification of software requirements by analogical reuse and second, to enable the integration of requirements specifications, generated by the multiple agents involved in information system development. Details of these applications can be found in sited references. Also, we have conducted an empirical evaluation of: (i) the consistency of the estimates generated by the model against human intuition about similarity and (ii) its recall performance in tasks of analogical retrieval, the results of which are presented in this paper.

1. Introduction

The remarkable ability of humans to understand novel situations by analogy to familiar ones and to solve new problems by remembering solutions to old analogous ones, has motivated the study of analogy as a non deductive paradigm of reasoning and its application in a variety of domains including law[4], medicine[16], economics[3], mechanical[59]) and software engineering[44,53,54,64,75,76,77].

Analogy has been studied in cognitive science, psychology, philosophy and artificial intelligence, yet from different perspectives. Cognitive scientists and psychologists are primarily concerned with how humans recall analogs and reason by analogy[22,24,47,56]. Philosophers concentrate on prerequisite conditions for drawing valid analogical inferences[38,46]. Finally, AI researchers focus on the development of computational models and systems for reasoning by analogy[26,31].

Along the latter direction of research, most of the computational models of analogical reasoning were developed as models of human cognition[20,28,47,65] without emphasizing efficiency and usability aspects, during the eighties. As a result, these models were very sensitive to the representation, the kinds and the domains of the analogs involved; highly dependent to the specific purpose of reasoning; operational only when informed by extensive amounts of knowledge determining critical aspects for the detection of analogies; and, computationaly expensive. As such, they could hardly support efficient analogical reasoning, applicable to complex tasks in different application domains.

To address these pragmatic limitations, our research focused on the development of a computational model for detecting analogies, which would:

(1) employ domain independent matching criteria;

(2) require no special, causal knowledge for determining analogies[51];

(3) be operational upon non uniformly represented analogs(i.e. analogs described via different sets of features and/or relations);

(4) be relatively tolerant to incomplete descriptions of analogs; and,

(5) scale efficiently along with increments in analogs complexity.

The developed similarity model operates on objects described using three *semantic modelling abstractions*, namely the *classification*, *generalization* and *attribution* [6,27,35,40,49]. The semantics of these abstractions have been taken into account in defining criteria for (1) deciding whether or not descriptional elements of two objects might be analogous to each other, and (2) discriminating the importance of such elements to the existence of analogies.

Analogies are detected by metric functions, which measure partial conceptual distances between objects with respect to their classification, generalization relations and attributes. These partial distances are aggregated in an overall object distance, which is transformed into a similarity measure indicating the aptness of detected analogies. None of the employed metrics uses any a-priori defined or user-supplied distance measures, which could limit the applicability of the model in tasks and domains where such measures wouldn't be readily available. The importance of the different classification, generalization relations and the attributes of objects to the existence of analogies is measured and taken into account while evaluating their partial distances. Importance measures are obtained by two functions, namely the *specialization depth* in the case of classification and generalization relations and the *salience* function in the case of attributes.

The selected representation framework is cognitively close to the human way of expressing knowledge[18] and has been utilized for constructing *conceptual models*(i.e. human-oriented descriptions) of artifacts in various application domains, including programming[15], software design[79] and cultural information systems[10]. This is because it can accommodate definitions of meta-models composed of types of concepts and relations meaningful in different domains and subsequently descriptions of artifacts as instances of these meta-models.

Having defined the similarity model over this framework, in a way requiring no domain-specific information for the detection of analogies, we can employ it for elaborating analogies between different sorts of artifacts in various application domains. So far the model has been applied to software engineering tasks. The first application concerns the modelling and elicitation of requirements specifications for software systems by reusing analogous specifications of existing systems[53,54] and the second the integration of specifications of software systems built in a distributed fashion[75,76]. These applications indicated an acceptable behaviour of the model regarding the pragmatic concerns motivated our research. Yet, a detailed discussion of them is beyond the scope of this paper and can be found in the sited references. Currently, we are investigating another application of the model, in particular the detection of analogies between traced enactments of software process models to intelligently guide humans in performing relevant activities[78].

The rest of this paper focuses on the formal definition and analysis of the model and is organized as follows. Section 2 describes the representation framework of the model, defines its distance and importance measuring functions and discusses the basic properties of these functions. Section 3 describes the computation of the model's functions and analyses its complexity. Section 4 presents an empirical evaluation of the model and section 5 compares it with other models for analogical reasoning. Finally, section 6 summarizes the model and discusses its open research issues. The paper has two appendices. The first gives an axiomatization of metrics and Dempster-Shafer belief functions, which underlie the definitions of the distance and the importance measuring functions of the model. The second includes proofs of theorems expressing basic properties of the model, which are presented in the main body of the paper, built upon these axiomatizations.

2. Formal Definition and Analysis of the Similarity Model

2.1 The Conceptual Modelling Framework

i) General description

The similarity model assumes a representation framework, where analogs are described using attributes, classification and generalization relations.

In this framework, entities and relationships are treated uniformly as objects with equal rights. Objects may belong to one or more classes introducing different kinds of attributes, which are used for building up object descriptions. Classification relations have a *set-membership* semantics. Classes themselves are objects, which are classified under metaclasses, metaclasses are objects classified under metametaclasses and so on. Furthermore, classes(but not individual objects) can be related through multiple generalization relations(i.e. Isa relations), provided that they have the same classification level (i.e. both are classes or metaclasses or metametaclasses and so on). Isa relations have a *set-inclusion* semantics and enforce the strict inheritance of attributes from superclasses to subclasses.

The treatment of attributes as objects, which can be grouped into classes and may have their own attributes, allows the definition of different kinds of relations, without the need of supplying specific representation primitives for each of them. Furthermore, the ability to introduce multiple and meta classification relations enables the definition of different models for describing objects.

In the following, we define - in a set theoretic manner - basic concepts of this framework, which are necessary for the subsequent treatment of the similarity model.

ii) Basic Definitions

The objects in the prescribed conceptual modelling framework are formally defined as:

Definition 1: *An object is a 7-tuple* $O_{id} = [id\ , l\ , FROM\ , In\ , Isa\ , A\ , TO\ ]$ *where*

id *is the object identifier of* O_{id} $(id\ \varepsilon\ I)$

l *is an object logical name of* O_{id} $(l\ \varepsilon\ L)$

FROM *is an object identifier denoting the possessing object of* O_{id} $(FROM\ \varepsilon\ I)$

In *is a finite set of object identifiers denoting the classes of* O_{id} $(In \subset I)$

Isa *is a finite set of object identifiers denoting the superclasses of* O_{id} $(Isa \subset I)$

A *is a finite set of object identifiers denoting the direct attributes of* O_{id} $(A \subset I)$

TO *is an object identifier denoting the value object of* O_{id} $(TO\ \varepsilon\ I)$

I *is a countably infinite set of symbols called object identifiers*

L *is a countably infinite set of symbols called object logical names*

Objects are uniquely identified by identifiers, due to the existence of an isomorphism *o*, which is internally constructed by enumeration as a by-product of their creation and is defined as:

Definition 2: *o is a total and onto isomorphism between the set of object identifiers I and the set of all the objects of an object base T,* $o : I \rightarrow T$ *such that forall* $x_1\ \varepsilon\ I$ *and* $x_2\ \varepsilon\ T$ *:*

$o(x_1) = x_2 \Longleftrightarrow x_2.id = x_1$

Also, objects are associated with logical names assigned to them directly by their creators. This association is implicitly defined, through the identifiers of objects, as:

Definition 3: *n is an onto, M:1 mapping between object identifiers and object logical names is* $n : I \rightarrow L$.

Objects are partitioned according to two criteria. The first of them concerns whether they represent entities or relationships in the real world and the second whether they represent atomic concepts, groups of atomic concepts, groups of groups of atomic concepts and so on. These criteria lead to the following two partitions, respectively:

i) the partition of objects into the set of individuals IU and the set of attributes AU ($IU \cap AU = \emptyset$); and,

ii) the partition of objects into the successive instantiation levels of *tokens*(i.e. the set C_0), *simple classes* (i.e. the set C_1), *meta classes*(i.e. the set C_2), *meta meta classes*(i.e. the set C_3) and so on ($\bigcap_i C_i = \emptyset \;\; i = 0,1,2, \cdots$).

The combination of the prescribed criteria results in another partition consisting of the sets IUC_i and AUC_i, defined as:

$$IUC_i = IU \cap C_i \;,\; AUC_i = AU \cap C_i \;\;,\;\; i = 0,1,2, \cdots$$

Class objects have an extension, including all their instances, defined as:

Definition 4: *The extension of a class object with identifier #i is a set of object identifiers, defined as:*

$$EXT[\#i] = \left\{ x \mid (\#i \;\varepsilon\; o(x).In \right\}$$

Also, we define the *shared extension* of a class as the set of its instances, which are also instances of at least one of its subclasses:

Definition 5: *The shared extension of a class with identifier #i* $EXT_s[\#i]$ *is defined as:*

$$EXT_s[\#i] = \left\{ x \;\mid\; (\#i \;\varepsilon\; o(x).In) \text{ and } (\;\exists\; y : (\#i \;\varepsilon\; o(y).Isa) \text{ and } (y \;\varepsilon\; o(x).In)) \right\}$$

Objects have attributes, which are said to belong to their *intensions*. Intensions of classes comprise their own attributes and the attributes inherited from their superclasses. Intensions are defined as follows:

Definition 6: *The intension of an object with identifier #i, INT[#i], is a set of attribute object identifiers, defined as:*

$$INT[\#i] = \begin{cases} o(\#i).A & \text{if } (\#i \;\varepsilon\; EXT[\#C0]) \\ o(\#i).A \cup INH[\#i] & \textit{otherwise} \end{cases}$$

where INH[#i] is the set of the inherited attributes of #i, defined as:

$$INH[\#i] = \Big\{ x \mid (\exists\, x_1 : (x_1 \,\varepsilon\, I) \text{ and } (x_1 \,\varepsilon\, o(\#i).Isa) \text{ and } (x \,\varepsilon\, o(x_1).A) \text{ and }$$

$$(not(\exists\, x_2 : (x_2 \,\varepsilon\, I) \text{ and } (x_2 \,\varepsilon\, o(\#i).A) \text{ and } (x \,\varepsilon\, o(x_2).Isa) \text{ and } (n(x) = n(x_2))) \text{ and }$$

$$(not(\exists\, x_3, x_4 : (x_3 \,\varepsilon\, I) \text{ and } (x_4 \,\varepsilon\, I) \text{ and } (x_3 \,\varepsilon\, o(\#i).Isa) \text{ and } (x_1 \,\varepsilon\, o(x_3).Isa) \text{ and }$$

$$(x_4 \,\varepsilon\, o(x_3).A) \text{ and } (x \,\varepsilon\, o(x_4).Isa) \text{ and } (n(x_4) = n(x))))) \Big\}$$

In reference to the modelling of attribute classes, we also define:

i) their *original class*(i.e. their most general superclass which has an identical logical name with them):

Definition 7: *The original class of an attribute class with identifier #i* $OC_{\#i}$ *is a class u satisfying the following condition*

$$(u \,\varepsilon\, o(\#i).Isa) \text{ and}((n(\#i) = n(u)) \text{ and } (not(\exists\, x_1 : (x_1 \,\varepsilon\, o(\#i).Isa) \text{ and } (x_1 \,\varepsilon\, o(u).Isa) \text{ and } (n(\#i) = n(x_1)))))$$

(ii) their *original domain class*(i.e. the class which first introduces attribute classes in a conceptual schema):

Definition 8: *The original domain class of an attribute class with identifier #i,* $ODC_{\#i}$ *is the domain class of its original class*

$$ODC_{\#i} = o(OC_{\#i}).FROM$$

(iii) their *scope*(i.e. the set of classes which an attribute class applies to):

Definition 9: *The scope of an attribute class with identifier #i, S[#i], is defined as:*

$$S[\#i] = \Big\{ x \mid ODC_{\#i} \,\varepsilon\, o(x).Isa \Big\}$$

(iv) their *refining classes* (these are the classes which either introduce or refine an attribute class):

Definition 10: *The set of the refining classes of an attribute with identifier #i, R[#i], is defined as:*

$$R[\#i] = \Big\{ c \mid (c \,\varepsilon\, S[\#i]) \text{ and } (\exists\, x_1 : (x_1 \,\varepsilon\, o(c).A) \text{ and } (n(\#i) = n(x_1))) \Big\}$$

(v) their *possible ranges*(i.e. the distinct classes which are used as class-ranges for an attribute class in a conceptual schema):

Definition 11: *The set of the possible attribute ranges of an attribute #i, AR[#i], is defined as:*

$$AR[\#i] = \left\{ x \mid (\exists\ x_1, x_2 : (x_1\ \varepsilon\ R[\#i]) \text{ and } (x_2\ \varepsilon\ o(x_1).A) \text{ and } (n(x_2) = n(\#i)) \text{ and } (o(x_2).TO = x)) \right\}$$

This conceptual modelling framework has been implemented as an object-oriented knowledge representation language called *Telos*, which is described in detail in[35,40].

2.2 The Distance and Similarity Functions

To avoid the possibility of comparing objects having different levels of abstraction in the prescribed conceptual modelling framework, similarity analysis obeys a basic ontological restriction, namely the principle of *Ontological Uniformity*. According to it, only objects which have the same instantiation level(i.e. they are both tokens or simple classes or meta classes or meta meta classes and so on) and furthermore both denote individuals or attributes can be compared with each other.

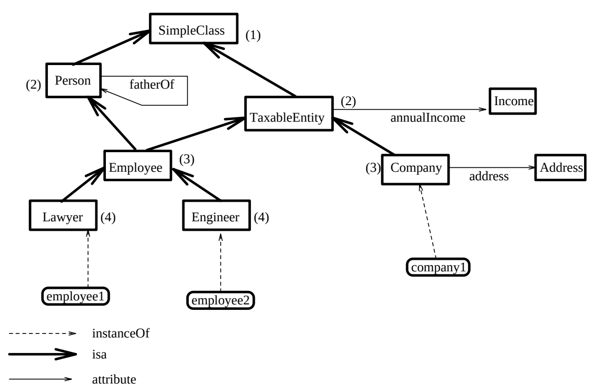

Figure 1: Ontologically Uniform and Not Uniform Objects

Thus, the similarity comparison between the attribute class *annualIncome* and the individual class *Person* as well as between the simple class *Person* and the token *company1* in figure 1 wouldn't be allowed. On the other hand, comparisons between objects classified under different classes of the same instantiation level, such as the objects *employee1* and *company1* in figure 1 are permitted. Hence, the ontological uniformity does not preclude the detection of analogies between objects with classification differences, such as the analogy in paying taxes regarding the objects *employee1* and *company1*. However, it considers unlikely the modelling of analogous objects at different instantiation levels and consequently precludes searching for analogies between such objects as meaningless.

Comparisons are based on distance functions. Distance functions, unlike other forms of matching such as the *ratio* and the *contrast* models of similarity [52,58,67,72], have well-defined mathematical properties (e.g. symmetry, triangularity) and intuitive geometric interpretations. In particular, ontologically uniform objects are compared by four partial distance functions, namely the *identification distance* (d_1), the *classification distance*(d_2), the *generalization distance* (d_3) and the *attribution distance* (d_4). These functions compare objects at different levels of detail and complement each other in preserving the ability of the model to detect analogies from descriptions of objects, which may be incomplete with respect to any of the prescribed relations.

2.2.1 The Identification Distance

The identification distance is introduced to provide a coarse-grain distinction between different objects with identical semantics(e.g. the two copies of the same book in figure 2) and is defined as follows:

Definition 12 : *The identification distance d_1 between two objects with identifiers #i , #j is defined as a function:* $d_1 : (\bigcup_{i=0}^{\infty}(IUC_i \times IUC_i)) \bigcup (\bigcup_{i=0}^{\infty}(AUC_i \times AUC_i)) \rightarrow [0,...,1]$

such that $d_1(\#i,\#j) = \begin{cases} 1 & \text{if } \#i \neq \#j \\ 0 & \text{if } \#i = \#j \end{cases}$

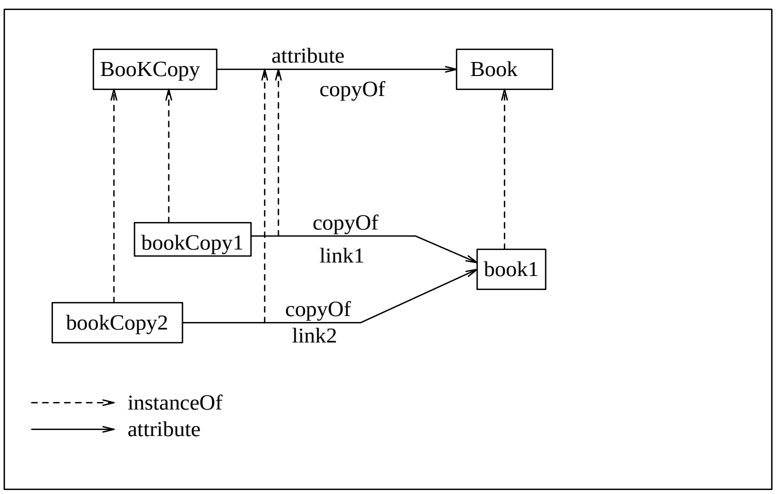

Figure 2: Non Identical Objects with Identical Semantics

2.2.2 The Classification Distance

The classification distance is introduced to give a rough estimate about the analogy over the substance of two objects by measuring and aggregating the importance of their non common classes. Formally, it is defined as follows:

Definition 13 : *The normalized classification distance d_2 between two objects with identifiers #i , #j is defined by the homographic transformation:*

$$d_2(\#i,\#j) = \frac{\beta_2 D_2(\#i,\#j)}{\beta_2 D_2(\#i,\#j) + 1}$$

D_2 is their absolute classification distance defined as:

$$D_2 : (\bigcup_{k=0}^{\infty}(IUC_i \times IUC_i)) \cup (\bigcup_{k=0}^{\infty}(AUC_i \times AUC_i)) \rightarrow [0,...,\infty) \quad D_2(\#i,\#j) = \sum_{x \in SDC[\#i,\#j]} \frac{1}{SD(x)}$$

where $SDC[\#i,\#j] = \left\{ o(\#i).In - o(\#j).In \right\} \cup \left\{ o(\#j).In - o(\#i).In \right\}$ *and,*

SD(x) is the maximum length of the paths connecting x with the most general class of its generalization taxonomy, called the specialization depth of x

D_2 measures the importance of classes by the inverse of their specialization depths. Classes with high specialization depths (i.e. classes placed at lower levels in generalization hierarchies) tend to express fine grain distinctions between homogeneous populations of objects, while classes with low specialization depths (i.e. classes placed at higher levels in generalization hierarchies) tend to express basic semantic distinctions between different types of objects. This weighting coincides with cognitive studies of classification in human mental models, which indicate that the more general a class the more significant the classification it express in a taxonomy [80]. According to definition 13, the absolute classification distances between the object *employee1* and the objects *employee2* and *company1* in figure 1 are 0.5(i.e. 1/4 + 1/4) and 1.41(i.e. 1/2 + 1/3 + 1/4 + 1/3), respectively.

The parameter β_2 in definition 13 determines the rate of asymptotic convergence of d_2 measures to 1 as D_2 goes to infinity. β_2 may evaluated so that d_2 be equal to 0.5 when D_2 takes its average value given a particular set of objects, thus making the estimation of the classification distance context-sensitive[58].

Function d_2 is a pseudometric:

Theorem 1: *The function* d_2 is a pseudometric.

In other words, it takes positive values, is triangular and symmetric. Also, it might take zero values even for non identical objects. Theorem 1, like all the theorems in sections 2 and 3, is proved in the second appendix of the paper.

2.2.3 The Generalization Distance

The generalization distance gives a rough estimate over the semantic differences of two classes, as evidenced by measuring and aggregating the importance of their non-common superclasses. Since token objects can not participate generalization relations this distance function is defined only between class objects:

Definition 14: *The normalized generalization distance* d_3 *between two class objects with identifiers #i , #j is defined as:*

$$d_3(\#i,\#j) = \begin{cases} \dfrac{\beta_3 D_3(\#i,\#j)}{1+\beta_3 D_3(\#i,\#j)} & \text{if } \#i,\#j \ \varepsilon\ IUC_i \quad i = 1,2,\cdots \\ d_o(\#i,\#j) & \text{if } \#i,\#j \ \varepsilon\ AUC_i \quad i = 1,2,\cdots \end{cases}$$

d_o *is the generalization distance between attribute classes defined as:*

$$d_o(\#i,\#j) = \begin{cases} 1 & \text{if } OC_{\#i} \neq OC_{\#j} \\ 0 & \text{if } OC_{\#i} = OC_{\#j} \end{cases}$$

D_3 *is their absolute generalization distance defined as:*

$$D_3 : (\bigcup_{i=1}^{\infty}(IUC_i \times IUC_i)) \rightarrow [0,\ldots,\infty) \quad D_3(\#i,\#j) = \sum_{x\ \varepsilon\ SDSC[\#i,\#j]} \frac{1}{SD(x)}$$

where $SDSC[\#i,\#j] = \left\{ o(\#i).Isa \cup (\{\#i\}) - o(\#j).Isa \cup \{\#j\} \right\} \cup \left\{ o(\#j).Isa \cup (\{\#j\}) - o(\#i).Isa \cup \{\#i\} \right\}$

SD(x) is the specialization depth of class x.

The generalization distance between individual classes is defined in a way similar to their classification distance except that their supeclasses are taken into account. Unlike it, the generalization distance between attribute classes depends on the identity of their original attribute classes.

The criterion of the original class identity in definition 14 enables a differentiation between refinements of inherited attribute classes (i.e. specialization of their range classes) representing the same relations or properties and specializations of attribute classes with shared but non identical semantics, both expressible by Isa relations in conceptual schemas. Consider for instance the specialization of the attribute class *identifiedBy* of *Person* by the attribute classes *identifiedBy* of *Soldier* and *hasSecurityNumber* of *Employee* in figure 3. The attribute class *identifiedBy* of *Soldier* refines the same identification relation modelled by the attribute class *identifiedBy* of *Person* for the particular case of soldiers(the latter is the original class of the former according to definition 7). On the other hand, the attribute class *hasSecurityNumber* of *Employee*, although a specialization of the class

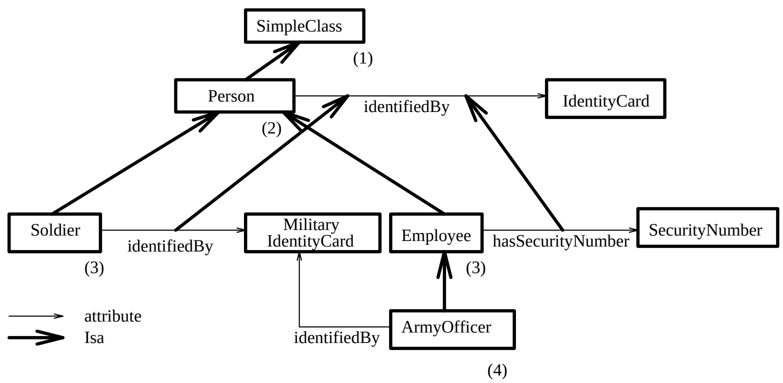

Figure 3: A Conceptual Schema of Classes of Persons

identifiedBy of *Person*, doesn't express the same relation with it. In fact, an employee may have an ordinal identification card in addition to his/her security number(notice that the attribute *identifiedBy* of *Person* is inherited by the class *Employee* according to definition 6). This difference is reflected by the generalization distances between the relevant pairs of attribute classes, which are 0 and 1 respectively.

Parameter β_3 has the same role and is estimated in the same way with parameter β_2(cf. section 2.2.2). Given its definition, the generalization distance is a metric:

Theorem 2: *Function* d_3 is a metric .

2.2.4 The Attribution Distance

The abstraction of attribution as defined in the modelling framework of similarity analysis is *semantically overloaded*[27]. In other words, attributes may be used for describing very different aspects of objects, including structural decompositions, non structural relations with other objects or even simple properties. This overloading may contribute to the expressive power of the framework but also introduces the problem of deciding whether arbitrary pairs of attributes are analogous while comparing their owning objects.

The similarity model deals with this problem by introducing the principle of the *semantic homogeneity*[61]. According to it, two attributes cannot be considered analogous unless they are semantically homogeneous. These by definition are attributes classified under exactly the same original attribute classes:

Definition 15: *Two attribute objects with identifiers #i , #j are semantically homogeneous* $sh(\#i, \#j)$*, if and only if:*

$$OCL[\#i] = OCL[\#j] \text{ where } OCL[x] = \left\{ x_1 \mid (x_1 = OC_{x_2}) \text{ and } (x_2 \, \varepsilon \, o(x).In) \right\} \; x = \#i \, , \#j$$

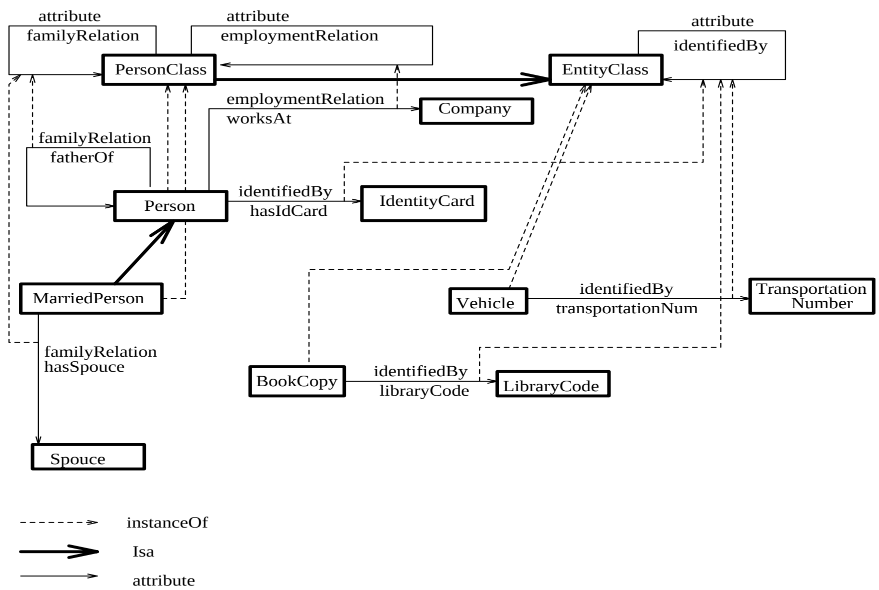

Figure 4: Semantically Homogeneous & Heterogeneous Attributes

---

In reference to figure 4, the attribute classes *libraryCode* of *BookCopy*, *transportationNum* of *Vehicle* and *hasIdCard* of *Person* are semantically homogeneous since they are all classified under the same original attribute metaclass *identifiedBy* of *EntityClass*. In fact, they have the same descriptive role for their owning objects, i.e. they uniquely identify them within the extensions of their classes. Consequently, they could be compared while detecting analogies between vehicles and copies of books or persons. Unlike them, the attribute class *worksAt* of *Person* is semantically heterogeneous with the attribute *hasSpouce* of *MarriedPerson*, since the former has been classified as an employment while the latter as a family relation.

As the following theorem states, semantic homogeneity is a reflexive, symmetric and transitive relation.

Theorem 3: *The semantic homogeneity of attributes is an equivalence relation*

The symmetry of semantic homogeneity ensures that attribute comparisons will be associative while its transitivity enables the definition of a metric over the attribution of objects(cf. proof of lemma 3 in appendix 2).

On the basis of semantic homogeneity, the set of *semantically homogeneous* pairs of attributes of two objects is defined as follows:

Definition 16: *The set of the semantically homogeneous of two objects with identifiers #i , #j* $SH[\#i,\#j]$ *is defined as:*

$$SH[\#i,\#j] = \left\{ (x_1,x_2) \mid (x_1 \,\varepsilon\, INT[\#i]) \text{ and } (x_2 \,\varepsilon\, INT[\#j]) \text{ and } sh(x_1,x_2) \right\}$$

The powerset of $SH[\#i,\#j]$ can be thought of as the set of all the possible interpretations of the analogy between #i and #j. Motivated by empirical evidence about the way humans interpret analogies[20], the similarity model considers only isomorphic mappings between the semantically homogeneous attributes of objects as valid interpretations of their analogy. The set of these valid mappings is formally defined as:

Definition 17: *The set of the valid isomorphisms between the semantically homogeneous attributes of two objects with identifiers #i , #j is defined as:*

$$C[\#i,\#j] = \left\{ c_k \mid (c_k \subseteq SH[\#i,\#j]) \text{ and } (\forall\; x_1, x_2 : ((x_1,x_2) \,\varepsilon\, c_k) \Rightarrow (not(\nexists\; x_3, x_4 : ((x_3, x_4) \,\varepsilon\, c_k) \text{ and } (((x_1 = x_3) \text{ and } (x_2 \neq x_4)) \; or \; ((x_1 \neq x_3) \text{ and } (x_2 = x_3))))) \text{ and } (not(\nexists\; c_{k'} : (c_{k'} \,\varepsilon\, C[\#i,\#j]) \text{ and } (c_{k'} \subset c_k)))) \right\}$$

Notice that since they are isomorphic, all the mappings in set C[#i,#j] are invertible. Therefore, the interpretation of the analogy between two objects using them can be grounded on exactly the same object elements, regardless of the direction of their comparison.

Each interpretation c_k determines the non analogous attributes of the involved objects(i.e. attributes not mapped by it):

Definition 18: *The set of the non analogous attributes of an object #i, with respect to an object #j, given their isomorphism* c_k is defined as:

$$A_{\#i}[c_k] = \left\{ x_1 \mid (x_1 \,\varepsilon\, INT[\#i]) \text{ and } (not(\nexists\; x_2, x_3 : ((x_2,x_3) \,\varepsilon\, c_k) \text{ and } (x_1 = x_2))) \right\}$$

Therefore, even semantically homogeneous attributes of objects may not be considered as being analogous, under different interpretations of their analogy.

From all the valid interpretations C[#i,#j] of the analogy between two objects, the similarity model selects the one with the minimum total distance. The total distance of each isomorphism c_k in C[#i,#j] is measured by the weighted sum of the distances between the pairs of attributes that constitute it and the importance of the non analogous attributes according to it. Thus, the selection of an interpretation for the analogy between two objects is based on an objective criterion of optimality, referred to as the principle of the *minimum distance isomorphism*. The total distance of the selected isomorphism between the attributes of two objects is defined as their attribution distance:

Definition 19: *The absolute attribution distance* D_4 *between two objects with identifiers #i , #j is defined as a function:*

$$D_4 : (\bigcup_{k=0}^{\infty}(IUC_i \ X \ IUC_i)) \bigcup (\bigcup_{k=0}^{\infty}(AUC_i \ X \ AUC_i)) \ X \ PowersetOf\,(IU \bigcup AU)) \rightarrow [0, \ldots, \infty)$$

$$D_4(\#i,\#j,V) = \begin{cases} \infty \ \text{ if } \ INT[\#i] = \varnothing \ or \ INT[\#j] = \varnothing \\ \min\limits_{c_k \ \varepsilon \ C[\#i,\#j]} \left\{ \sum\limits_{(x_1,x_2)\ \varepsilon\ c_k} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum\limits_{x_3 \ \varepsilon\ A_{\#i}[c_k]} s(x_3)^2 + \sum\limits_{x_4\ \varepsilon\ A_{\#j}[c_k]} s(x_4)^2 \right\} \quad otherwise \end{cases}$$

$$d'(\#i,\#j,V) = \frac{\beta D'(\#i,\#j,V)}{\beta D'(\#i,\#j,V)+1}$$

$$D'(\#i,\#j,V) = \begin{cases} (\underline{PD_{ec}}\,\underline{W_1}\,\underline{PD_{ec}}^T)^{1/2} \ \text{ if } \ (\#i, \#j \ \varepsilon \bigcup_{k=1}^{\infty} AUC_i) \text{ and } ((o(\#i).TO \ \varepsilon\ V) \ or \ (o(\#j).TO \ \varepsilon\ V) \\ (\underline{PD_{et}}\,\underline{W_2}\,\underline{PD_{et}}^T)^{1/2} \ \text{ if } \ \#i, \#j \ \varepsilon\ AUC_0 \text{ and } ((o(\#i).TO \ \varepsilon\ V) \ or \ (o(\#j).TO \ \varepsilon\ V) \\ D(\#i,\#j,V) \qquad otherwise \end{cases}$$

where

$$W_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 6 & 1 \\ 0 & 6 & 6 & 6 \\ 0 & 1 & 6 & 1 \end{bmatrix} \quad W_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 6 \\ 0 & 6 & 6 \end{bmatrix}$$

$$PD_{ec} = [d_1(\#i,\#j) \quad d_2(\#i,\#j) \quad d_3(\#i,\#j) \quad d_4(\#i,\#j)\,]$$

$$PD_{et} = [d_1(\#i,\#j) \quad d_2(\#i,\#j) \quad d_4(\#i,\#j,V \bigcup \{\#i,\#j\})\,]$$

V is a set of identifiers of objects($V \subseteq IU \bigcup AU$) and,

$$s(x_i) = \max_{x_j \ \varepsilon\ OCL[x_i]} \{SL(x_j)\} \quad SL(x_j) \ is \ the \ salience \ of \ \ x_j \,(see \ \ definition \ 32 \ below), s(x_i) \ \varepsilon \ [0,...,1]$$

D_4 selects minimum distance isomorphic mappings between the attributes at all the successive levels in the transitive closures of the attribution graphs of the involved objects, recursively (see figure 5). Also, it assigns an infinum distance measure to pairs of objects, whose intensions are empty. In this case, it essentially assumes that the yet unknown intensions of objects will more likely include semantically heterogeneous attributes rather than homogeneous ones.

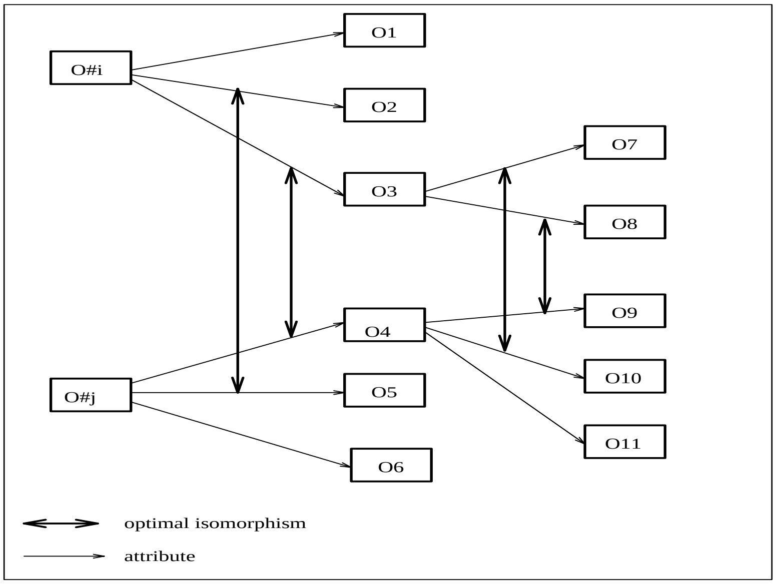

Figure 5: Synthesis of Successive Optimal Interpretations of Analogies

In cases of attributes having as values objects, whose distance to some other object is currently being estimated (i.e. the members of set V in definition 19), recursion is controlled by estimating their distance only with respect to their identification, classification, generalization and attribution but not with respect to these values.

The absolute distance measures D_4 are normalized into relative ones, according to the following definition:

Definition 20: *The normalized attribution distance d_4 between two objects with identifiers #i , #j is defined as a function:*

$$d_4(\#i,\#j) = \frac{\beta_4 D_4(\#i,\#j)}{\beta_4 D_4(\#i,\#j) + 1}$$

Theorem 4: *Function d_4* is a metric.

An implication of the definitions of the attribution and generalization distances is that the pairs of the semantically homogeneous attributes of two objects, which also have the same original class, will always belong to their optimal isomorphism:

Theorem 5: *All the pairs of attributes (x_1, x_2) of two objects #i, #j that belong to the set SH[#i,#j] and have the same original class belong to the optimal isomorphism c_{opt} between the objects #i and #j.*

This property is reasonable considering that attributes sharing of a common original class express the same relation, as discussed in section 2.2.3.

2.2.5 The Overall Distance Function

The partial metrics d_1,d_2,d_3 and d_4 are aggregated into an overall distance D according to the following quadric function:

Definition 21: *The overall distance* D *between two objects, identified by #i,#j, is defined as a function:*

$$D : \left(\bigcup_{k=0}^{\infty}(IUC_i \ X \ IUC_i)\right) \bigcup \left(\bigcup_{k=0}^{\infty}(AUC_i \ X \ AUC_i)\right) \ x \ PowersetOf \ (IU \bigcup AU) \rightarrow [0,...,\infty)$$

$$D(\#i,\#j,V) = \begin{cases} (\underline{PD}_{ec}\,\underline{W}_{ec}\,\underline{PD}_{ec}^{\,T})^{1/2} \ \text{ if } \ \#i\,,\#j \ \varepsilon \bigcup_{k=1}^{\infty} IUC_i \\ (\underline{PD}_{et}\,\underline{W}_{et}\,\underline{PD}_{et}^{\,T})^{1/2} \ \text{ if } \ \#i\,,\#j \ \varepsilon \ IUC_0 \\ (\underline{PD}_{ac}\,\underline{W}_{ac}\,\underline{PD}_{ac}^{\,T})^{1/2} \ \text{ if } \ \#i\,,\#j \ \varepsilon \bigcup_{k=1}^{\infty} AUC_i \\ (\underline{PD}_{at}\,\underline{W}_{at}\,\underline{PD}_{at}^{\,T})^{1/2} \ \text{ if } \ \#i\,,\#j \ \varepsilon \ AUC_0 \end{cases}$$

where

$$PD_{ec} = [d_1(\#i,\#j) \quad d_2(\#i,\#j) \quad d_3(\#i,\#j) \quad d_4(\#i,\#j)\ ]$$

$$PD_{et} = [d_1(\#i,\#j) \quad d_2(\#i,\#j) \qquad d_4(\#i,\#j,V \bigcup \{\#i,\#j\})\ ]$$

$$PD_{ac} = [d_1(\#i,\#j) \quad d_2(\#i,\#j) \quad d_3(\#i,\#j) \quad d_4(\#i,\#j,V \bigcup \{\#i,\#j\}) \quad D(o(\#i).TO,o(\#j).TO,V \bigcup \{\#i,\#j\})\ ]$$

$$PD_{at} = [d_1(\#i,\#j) \quad d_2(\#i,\#j) \qquad d_4(\#i,\#j,V \bigcup \{\#i,\#j\}) \quad D(o(\#i).TO,o(\#j).TO,V \bigcup \{\#i,\#j\})\ ]$$

$$\underline{W}_{ec} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \quad \underline{W}_{et} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \quad \underline{W}_{ac} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 6 & 1 & 0 \\ 0 & 6 & 6 & 6 & 0 \\ 0 & 1 & 6 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \underline{W}_{at} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A series of experiments, whose results are summarized by table 1 indicated statistically significant correlations between classification/generalization and attribution distances, thus prompting the use of a quadric functional form for the overall distance D. These experiments were performed using conceptual models of requirements specifications (cases 1,2,3,4), abstractions of domain knowledge for information systems[44](case 5), static analysis data of CooL programs[70], C++ programming constructs [15](cases 7,8) and cultural artifacts[10](cases 9,10). In 4 out of the 6 cases where both d_2 and d_3 were definable(i.e. the involved objects were not tokens) and not equal to 0, a statistically significant($p < 0.05$) positive correlation between their resulting measures was found(cf. coefficients marked with asterisks in table 1).

Table 1: Correlation Coefficients Between Partial Distances				
Case	Set Size	d_2, d_3	d_2, d_4	d_3, d_4
1	231	-	-	.81*
2	28	.47*	.59*	.6*
3	6	0	.74	.08
4	861	.45*	.12*	.51*
5	10	-	-	.53
6	903	-	.15*	-
7	171	.24*	.09	.64*
8	66	-0.05	.17	.21
9	741	-	-	.39*
10	946	.55*	.6*	.4*

As a quadric aggregate of distance metrics, the overall object distance D is a metric itself:

Theorem 6: *Function D is a metric.*

2.2.6 The Similarity Function

The overall distance measures between objects are transformed into similarity measures, according to the following function:

Definition 22: *The similarity* S *between two objects with identifiers #i,#j is defined as a function:*

$$S : \left(\bigcup_{k=0}^{\infty}(IUC_i \ X \ IUC_i)\right) \bigcup \left(\bigcup_{k=0}^{\infty}(AUC_i \ X \ AUC_i)\right) \ x \ powersetOf \ (IU \bigcup AU) \rightarrow [0,...,1]$$

$$S(\#i,\#j,V) = e^{-N*D(\#i,\#j,V)} \ , N \ \varepsilon \ (0, \cdots, \infty)$$

Similarity measures indicate the aptness of the analogy between two objects.

2.2.7 An Example of Similarity Analysis

In the following, we give an example of using the prescribed functions in estimating the conceptual distances and thereby detecting the analogy between the classes *LibraryBorrower* and *CarAgencyCustomer*, given the conceptual model of figure 6. The partial and the overall distances as well as the similarity measures between the objects in this model are shown in table 2(these measures were obtained by setting all the $s(x_i)$ measures in definition 19 equal to 1).

Library borrowers and car agency customers do not have any non-common classes(they are both instances of the meta-class *EntityClass*) and therefore they have a zero classification distance(cf. column d_2 in table 2). Also, they are both specializations of the same superclass, i.e. *ResourceBorrower*, thus having a relatively low generalization distance, that is 0.56(cf. column d_3 in table 2).

The computation of their attribution distance results into an isomorphism between their attributes, depicted by the thick dotted lines in figure 6. This isomorphism maps the attributes *libraryCard* and

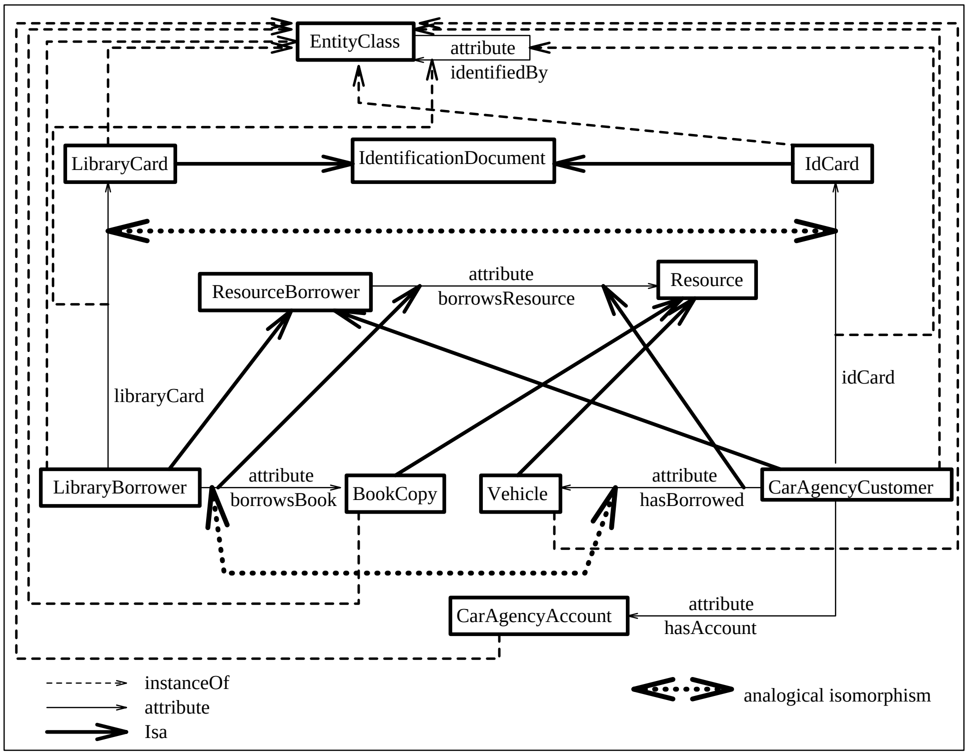

Figure 6: Analogy Between Library Borrowers and Car Agency Customers

borrowsBook of *LibraryBorrower* onto the attributes *idCard* and *hasBorrowed* of *CarAgencyCustomer*, respectively. In fact, these pairs of attributes have analogous roles for their owning classes.

The attributes of the first pair enable the identification of library borrowers and car agency customers as borrowers. Such an identification is possible through a card awarded by the library in the case of library borrowers and through their ordinary identification cards in the case of car agency customers. This dissimilarity does not devalidate the underlying analogy evidenced from the conceptual schema through the classification of the relevant attributes under the attribute class *identifiedBy* of *EntityClass*. On the contrary, the analogy becomes more evident by the fact that the object-values of these attributes(i.e. the classes *LibraryCard* and *IdCard*) are specializations of a common superclass, namely the class *IdentificationDocument*, which decreases their generalization distance (since it increases their specialization depths). Notice that the mapping of the attribute *libraryCard* onto the attribute *idCard* was the only valid one according to the criterion of the semantic homogeneity.

Table 2: Distance Measures Between Objects of Figure 6

Objects	d_1	d_2	d_3	d_4	D_{TO}	D	S
		$\beta_2 = 2$	$\beta_3 = 2$	$\beta_4 = .5$			N=0.3
LibraryBorrower,CarAgencyCustomer	1	0	0.56	0.34	-	1.61	0.61
BookCopy,Vehicle	1	0	0.56	1	-	1.36	0.66
LibraryCard,Vehicle	1	0	0.56	1	-	1.36	0.66
BookCopy,CarAgencyAccount	1	0	0.72	1	-	1.79	0.58
LibraryCard,CarAgencyAccount	1	0	0.72	1	-	1.79	0.58
borrowsBook,hasBorrowed	1	0	1	1	1.36	2.2	0.51
libraryCard,idCard	1	0	1	1	1.36	2.2	0.51
borrowsBook,hasAccount	1	0.5	1	1	1.79	2.33	0.49

Unlike it, the mapping of the attribute *borrowsBook* onto the attribute *hasBorrowed* was not the only valid one, according to the same criterion. In fact, since *borrowsBook* was also semantically homogeneous to the attribute *hasAccount* of *CarAgencyCustomer*(they were both instances of the attribute meta-class *attribute*, whose extension corresponds to set *AU* in section 2.1) it could have been mapped on it, alternatively. However, as it can be observed from table 2, the overall distance between the attributes *borrowsBook* and *hasBorrowed* is less than the overall distance between *borrowsBook* and *hasAccount*(2.2 vs. 2.33). This is due to the distances of their values: the overall distance *D* between *BookCopy* and *Vehicle* equals 1.36 while the same distance between *BookCopy* and *CarAgencySAccouny* equals 1.79. Consequently, *borrowsBook* was mapped onto *hasBorrowed* due to the criterion of the minimun distance isomorphism.

2.3 The Salience Functions

2.3.1 The Problem of Salience

Like classification and generalization relations, different attributes are expected to have varying degrees of importance to the existence of analogies, depending on their role to the description of their owning objects[47]. Usually, attributes expressing fundamental structural and behavioural characteristics of objects are more important to the existence of analogies than attributes which express simple properties.

Often, computational models of analogical reasoning make such distinctions of importance on the basis of:

(1) syntactic differences regarding the representation of attributes(e.g. SME[17]);

(2) direct estimates provided by users(e.g. ARCS[65],MACKBETH[72]);

(3) assessments about the results of reasoning sessions provided by external observers(e.g. CBL4[2],CBR+EBL[14]); and

(4) knowledge determining the attributes, which are related to the purpose of reasoning(expressed either by causal relations between attributes and goals as in Protos[48] and PRODIGY[69] or by indices to descriptions of analogs in memory as in MEDIATOR[37]).

Knowledge-based distinctions are usually bounded by the knowledge acquisition bottleneck. On the other hand, user-based ones might be sensitive to subjective biases. As opposed to them, the similarity model estimates the importance of attributes based on the concept of *attribute dominance*. Attribute dominance is introduced as a compound property derived from three primitive properties of attributes, namely their *charactericity*, *abstractness* and *determinance*. These are defined by logical conditions on the representation of attributes in conceptual models. Salience is then introduced as belief that an attribute is dominant and provides a graded alternative to the logical strictness of dominance.

2.3.2 The Properties Underlying Attribute Dominance

i) The Charactericity of Attributes

We distinguish as characteristic attributes which discriminate the different classes they apply to in a conceptual schema. This discrimination can be evidenced from the refinement of attributes (i.e. the specialization of their range-classes) by classes which inherit them in generalization taxonomies. Such refinements express how subclasses differ from their superclasses. Attributes refined in many classes of a generalization taxonomy become characteristic of it. Formally, the charactericity of an attribute is defined as:

Definition 23: *An attribute class with identifier #i is characteristic* $CH_{\#i}$ *if and only if* $|S[\#i]| = |AR[\#i]|$

For instance, the attribute *qualification* in figure 7 is characteristic since it has distinct range classes in all the classes of its scope thereby distinguishing the different subclasses of employees, according to definition 23. Unlike it, the attribute *worksFor*, which represents the relation between a researcher and the project he/she is involved in, is not characteristic since there exist employees not related with any project(e.g. administrative staff). Furthermore, the special types of researchers are not differentiated by the projects they work for.

ii) The Abstractness of Attributes

Attributes have different significance for the existence and behaviour of their objects, subject to the class which introduces them in a conceptual schema(i.e. their *original domain class*). Classes may be divided into *abstract* and *concrete*, according to whether or not they have instances of their own [1,42,71]. Concrete classes introduce attributes enabling a detailed description of their instances,

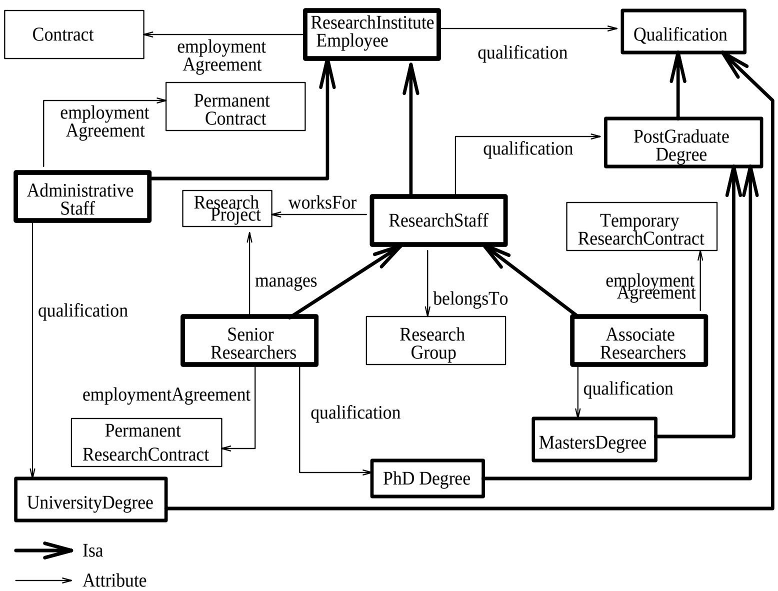

Figure 7: A Conceptual Schema of Employees of a Research Institute

while abstract ones introduce only attributes which are essential for the structure and the functional behaviour of the instances of their concrete subclasses[80]. Consider for example the difference in the significance of the attribute *hasSteeringSystem* of a vehicle and the attribute *luggageCarryingCapacity* of a car. The former is essential for driving any sort of vehicle(e.g. airplanes, trains, cars). Unlike it, the latter neither applies to all vehicles(e.g. fighting airplanes) nor relates to the functionality of the special kind of vehicles, it applies to. Thus, we introduce the abstractness of attributes as a property reflecting this behavioural and structural significance and define it on the basis of the abstractness of their original domain classes.

In particular, we define abstract classes as:

Definition 24: *A class with identifier #i is abstract* $AB_{\#i}$, *if and only if* $EXT[\#i]=EXT_s[\#i]$

and abstract attributes as:

Definition 25: *An attribute class with identifier #i is abstract (i.e.* $ABS_{\#i}$*) if and only if its original domain class #u(i.e.* $\#u = ODC_{\#i}$ *) is abstract (i.e.* $AB_{\#u}$*)*

According to these definitions, the attributes *qualification* and *employmentAgreement* in figure 7 are abstract since their common original domain class is the abstract class *ResearchInstituteEmployee*(we assume that it has no instances of its own). Unlike them, the attribute *manages*, whose original domain class is the concrete class *SeniorResearchers*, is not abstract.

iii) The Determinance of Attributes

We consider as *determinant* attributes whose values determine the values of other attributes. For example, the model of a car could dictate the place where it has been produced if the relevant manufacturer produces its different models in different countries. We restrict our interest in dependencies between attributes that have the same domain and in a special case of them, namely the *total equivalences*. Both these types of dependencies can be precluded for specific attributes, given their modelling in a conceptual schema.

Suppose that two attributes x and y are defined as associations $x : D_x \rightarrow I_x$, $y : D_y \rightarrow I_y$ where I_x and I_y denote their images(i.e. the sets of their actual values). In our view, the attribute y might depend on the attribute x(i.e. M_{xy}) only if they have identical domains, $D_x \equiv D_y$ (i.e. both of them apply to exactly the same set of objects). Given that they have, their dependency is defined as a mapping M between their values(i.e. $M : I_x \rightarrow I_y$). A total equivalence between x and y exists in cases where M is a total and onto isomorphism.

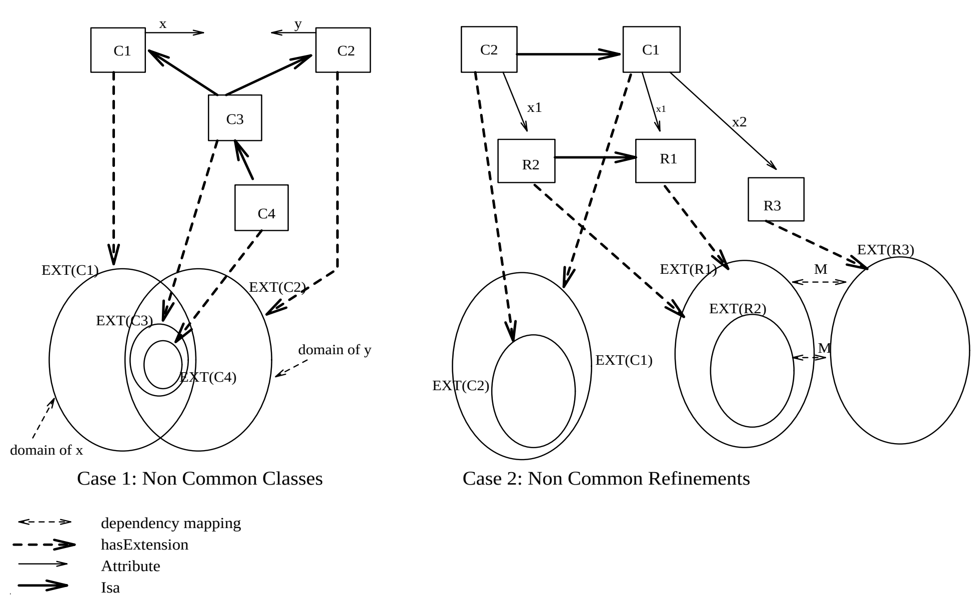

Figure 8: Non Common Classes and Refinements in Conceptual Schemas

The domain equality condition of dependencies is checked by the identity of the scopes of the involved attributes(i.e. $S[\#i] = S[\#j]$). Non common classes in the scopes of two attributes indicate that they have non identical domains. Consider for example the attributes x and y in case 1 of figure 8. The classes C1 and C2 in their scopes are not common to both of them. In fact, there must be objects belonging to the class C1 but not to C2 and vice versa(i.e. the sets EXT[C1]-EXT[C2] and EXT[C2]-EXT[C1] will be non-empty in general) regardless of the current state of their extensions. If this was not the case, there would be no need for maintaining both C1 and C2, two classes not related by a generalization relation, in the conceptual schema. Hence, we will consider dependencies between attributes x and y as not definable(i.e. $\overline{M}_{xy}$ and $\overline{M}_{yx}$) whenever:

$$(cnd1)\ :\ S[x] \neq S[y]$$

Also, total equivalences might exist subject to a second definability condition. Given two attributes x and y, it will be impossible to define a total equivalence mapping between their images whenever their refining classes do not coincide:

$$(cnd2)\ :\ ((S[x]-R[x]) \cap R[y])) \cup ((S[y]-R[y]) \cap R[x]) \neq \emptyset$$

Case 2 in figure 8 exemplifies this condition. The attribute x1 is refined by class C2 unlike the attribute x2, which is simply inherited by it. If an equivalence mapping(i.e. a total and onto isomorphism) between these two attributes were definable their images with respect to C1 and C2, should have equal numbers of elements(i.e. $|EXT(R1)| = |EXT(R3)|$ and $|EXT(R2)| = |EXT(R3)|$). However, since attribute x1 is refined in C2 its image with respect to C2 will have fewer elements than its image with respect to C1 (since R2 is a subset of R1 $|EXT(R2)| < |EXT(R1)|$) which is a contradiction.

Notice that, according to conditions *cnd1* and *cnd2* no dependency mapping could be defined between the attributes *worksFor* and *qualification* in figure 7. By contrast, a dependency between the attributes *worksFor* and *belongsTo* (it express the research group of a researcher) in the same figure might exist. These attributes have identical scopes and they are not refined separately. In fact, projects are normally allocated to research groups and thus the projects a researcher is involved with depend on his(her) group.

It must be pointed out that, failing to preclude the definability of a dependency mapping indicates that a dependency might exist but it doesn't necessarily imply it.

On the basis of interattribute dependencies we define determinative attributes as:

Definition 26: *An attribute class with identifier #i is determinative* $DET_{\#i}$, *if and only if* $(\exists\ x, y\ :(x\ \varepsilon\ S[\#i])$ and $(y\ \varepsilon\ INT[x])$ and $M_{iy})$

According to this definition, it is sufficient to identify even one dependency between an attribute x and another attribute y of a class in its scope for characterizing x as determinative.

iv) The Dominance of Attributes

We define the dominance of attributes on the basis of their charactericity, abstractness and determinance, as follows:

Definition 27: *An attribute class with identifier #i is dominant* $DOM_{\#i}$, *if and only if*

$(ABS_{\#i}$ and $CH_{\#i}$) *or* $DET_{\#i}$

The conjunction of abstractness and characterICITY excludes cases satisfying only one of those properties, as extreme ones. In fact, certain attribute classes may be introduced in abstract classes only because of their broad applicability(e.g. the attribute class *securityNumber* of *ResearchInstituteEmployee* in figure 9). Also, attribute classes, which are introduced in leaf classes of generalization taxonomies, although satisfy the definition of charactericity do not have a general classification significance for them(e.g. the attribute class *responsibleFor* of *AdministrationManager* in figure 9).

2.3.4 Attribute Salience: Belief on their Dominance

The logical conditions defining the charactericity, abstractness and determinance may prove too restrictive and be violated by attributes intuitively satisfying these properties(especially due to flaws in conceptual models). To overcome this problem, we derive inferences about the properties underlying attribute dominance(and consequently the dominance itself) in an approximate way, where their truth values are associated with evidence measures.

In particular, we introduce attribute *salience* as a measure of evidence that an attribute is dominant or not. Salience is estimated from evidence measures about the truth values for each of the underlying properties of dominance. These measures are obtained by evidence functions, which measure the extent to which the defining conditions of the relevant property are satisfied in a conceptual model and are interpreted as belief functions in the context of the Dempster-Shafer theory of evidence[57](since as we prove in appendix 2, they satisfy the axioms defining such beliefs).

i) A Frame of Discernment for Defining and Combining Evidence Functions

The definition and combination of evidence functions for the charactericity, abstractness and determinance is based on the introduction of a *frame of discernment* Θ_i [57] for each attribute #i.

Θ_i consists of proposition vectors $[vc, va, vm_1, vm_2, ..., vm_n]$, whose elements are variables indicating the truth value of the charactericity (i.e. variable vc) and the abstractness(i.e. variable va) of an attribute #i as well as the definability of a dependency mapping between #i and the *n* other attributes defined for the classes of its scope (variables vm_j). By convention, the value 1(0) for any of these variables means the truth(falsity) of the relevant property. Each proposition vector represents the conjunction of its elements by assumption. Θ_i includes all the proposition vectors which can be formed by taking all the possible combinations of values for these variables.

Given Θ_i, the properties determining the dominance of #i are defined as the following subsets of it:

$$CH_i \equiv \left\{ [vc, va, vm_1, vm_2, ..., vm_n] \mid vc = 1 \right\}$$

$$\overline{CH_i} \equiv \left\{ [vc, va, vm_1, vm_2, ..., vm_n] \mid vc = 0 \right\}$$

$$ABS_i \equiv \left\{ [vc, va, vm_1, vm_2, ..., vm_n] \mid va = 1 \right\}$$

$$\overline{ABS_i} \equiv \left\{ [vc, va, vm_1, vm_2, ..., vm_n] \mid va = 0 \right\}$$

$$DET_i \equiv \bigcup_{j=1}^{n} M_j \quad \text{where } M_j \equiv \left\{ [vc, va, vm_1, vm_2, ..., vm_n] \mid vm_j = 1 \right\}$$

$$\overline{DET_i} \equiv \bigcap_{j=1}^{n} \overline{M_j} \quad \text{where } \overline{M_j} \equiv \left\{ [vc, va, vm_1, vm_2, ..., vm_n] \mid vm_j = 0 \right\}$$

$$DOM_i \equiv (CH_i \cap ABS_i) \cup DET_i$$

$$\overline{DOM_i} \equiv (\overline{CH_i} \cup \overline{ABS_i}) \cap \overline{DET_i}$$

ii) The Evidence to Charactericity

Different specializations of classes in conceptual schemas are not always associated with refinements of the same attribute classes. For instance, the attribute *employementAgreement*, which express the contract of an employment, is not refined by the class *ResearchStaff* in figure 7. Nevertheless, it is characteristic of the other classes of employees in the relevant taxonomy. Such cases motivated the definition of an evidence measure about charactericity, as follows:

Definition 28: *The evidence to the charactericity of an attribute class with identifier #i,* m_i^{ch} *is measured by the function:*

$$m_i^{ch}(P) = \begin{cases} c_i & \text{if } P = CH_i \\ 1 - c_i & \text{if } P = \overline{CH_i} \\ 0 & \text{if } P \subseteq \Theta \;\&\; P \neq CH_i \;\&\; P \neq \overline{CH_i} \end{cases}$$

where $c_i = \dfrac{|AR[i]|}{|S[\#i]|}$

$m_i^{ch}(P)$ relaxes the logical definition of charactericity giving an evidence measure equal to 1 whenever this definition is satisfied ($|AR[\#i]| = |S[i]| \rightarrow c_i = 1$). It also satisfies the axiomatic definition of the Dempster-Shafer basic probability assignments(cf. appendix 1):

Theorem 7: *The evidence function $m_i^{ch}(P)$ is a basic probability assignment.*

iii)The Evidence to Abstractness

Since an abstract attribute is defined by consequence of the abstractness of its original domain class, the evidence about its own abstractness is also estimated from the evidence about the abstractness of this class:

Definition 29: *The evidence to the abstractness of an attribute class with identifier #i, m_i^a is measured by the function:*

$$m_i^a(P)=\begin{cases} e_j & \text{if } P = ABS_i \\ 1-e_j & \text{if } P = \overline{ABS_i} \\ 0 & \text{if } P \subseteq \Theta \;\&\; P \neq ABS_i \;\;\&\;\; P \neq \overline{ABS_i} \end{cases}$$

where $j = ODC_{\#i}$, $e_j = \dfrac{|EXT_s[\#j]|}{g(EXT[\#j])}$ *and*, $g(EXT[\#j]) = \begin{cases} |EXT[\#j]| & \text{if } EXT[\#j] \neq \emptyset \\ 1 & \text{if } EXT[\#j] = \emptyset \end{cases}$

$m_i^a(P)$ relaxes the logical definition of abstractness giving an evidence measure equal to 1, whenever this definition is satisfied. In fact, provided that $EXT[\#j] \neq \emptyset$, if $|EXT_s[\#j]| = |EXT[\#j]|$ then $e_j = 1$. $m_i^a(P)$ satisfies the axiomatic definition of the Dempster-Shafer probability assignments:

Theorem 8: *The function $m_i^a(P)$ is a basic probability assignment.*

iv) The Evidence to Undefinability of Dependencies

The evidence about the undefinability of a dependency mapping between two attributes #i and #j, due to their non common classes is measured by the following function:

Definition 30: *The evidence to the undefinability of a dependency mapping between an attribute class with identifier #i and an attribute class with identifier #j, due to their non common scope, m_{ij}^{ncs} , is measured by the function:*

$$m_{ij}^{ncs}(P)=\begin{cases} d_{ij} & \text{if } P = \overline{M}_j \\ 1-d_{ij} & \text{if } P = \Theta \\ 0 & \text{if } P \subseteq \Theta \;\&\; P \neq \overline{M}_j \;\&\; P \neq \Theta \end{cases}$$

where $d_{ij} = \dfrac{(|S[\#i]-S[\#j]|+|S[\#j]-S[\#i]|)}{(|S[\#i] \cup S[\#j]|)}$

According to $m_{ij}^{ncs}(P)$, the more the non common classes in the scopes of two attributes the more unlikely to be able to define a dependency mapping between them. $m_{ij}^{ncs}(P)$ satisfies the axiomatic definition of the Dempster-Shafer probability assignments:

Theorem 9: *The function $m_{ij}^{ncs}(P)$ is a basic probability assignment.*

v) The Evidence to Undefinability of Total Equivalences

The evidence to the undefinability of total equivalence mappings between two attributes #i and #j, due to their non common refinements over a conceptual schema, is measured according to the following definition:

Definition 31: *The evidence to the undefinability of a total equivalence mapping between an attribute class with identifier #i and an attribute class with identifier #j, due to their non common refinements over a conceptual schema,* m_{ij}^{ncr} *is measured by the function:*

$$m_{ij}^{ncr}(P)=\begin{cases} r_{ij} & \text{if } P=\overline{M}_j \\ 1-r_{ij} & \text{if } P=\Theta \\ 0 & \text{if } (P\subseteq\Theta \ \& \ P\neq\overline{M}_j \ \& \ (P\neq\Theta) \end{cases}$$

where $r_{ij}=\dfrac{|(S[\#i]-R[\#i])\cap R[\#j]|+|(S[\#j]-R[\#j])\cap R[\#i]|}{|R[\#i]\cup R[\#j]|}$

According to $m_{ij}^{ncr}(P)$, the more the non common refinements of two attributes over a schema, the more evident the impossibility of defining a total equivalence mapping between them. $m_{ij}^{ncr}(P)$ satisfies the axiomatic definition of the Dempster-Shafer probability assignments:

Theorem 10: *The function* $m_{ij}^{ncr}(P)$ *is a basic probability assignment.*

vi) Total Belief and Plausibility of Dominance

The total belief and the plausibility to the dominance of an attribute #i(i.e. $Bel(DOM_i)$ and $P^*(DOM_i)=1-Bel(DOM_i)$) are estimated by combining the basic probability assignments m_i^{ch} and m_i^{a} with the assignment m_{ij}^{ncs}, when every kind of interattribute dependency is taken into account:

Theorem 11: *The combination of the basic probability assignments* m_i^{a}, m_i^{ch} and m^{ncs} *results into the following Dempster-Shafer combined beliefs:*

$$Bel(DOM_i)=c_i e_u$$

$$Bel(\overline{DOM_i})=(1-c_i e_u)\prod_{j=1}^{n} d_{ij}$$

$$P^*(DOM_i)=1-(1-c_i e_u)\prod_{j=1}^{n} d_{ij}$$

where $u=ODC_{\#i}$.

On the other hand, these beliefs and plausibilities are estimated by combining the basic probability assignments m_i^{ch} and m_i^{a} with the assignments m_{ij}^{ncs} and m_{ij}^{ncr}, when only total equivalencies are taken into account:

Theorem 12: *The combination of the basic probability assignments* m_i^a, m_i^{ch} *with the basic probability assignments* m_{ij}^{ncs} and m_{ij}^{ncr} *results into the following Dempster- Shafer combined beliefs:*

$$Bel(DOM_i) = c_i e_u$$

$$Bel(\overline{DOM_i}) = (1 - c_i e_u) \prod_{j=1}^{n} (1 - (1 - r_{ij})(1 - d_{ij}))$$

$$P^*(DOM_i) = 1 - (1 - c_i e_u) \prod_{j=1}^{n} (1 - (1 - r_{ij})(1 - d_{ij}))$$

where $u = ODC_{\#i}$.

vii) The Salience Measuring Function

The salience of an attribute class is measured from the belief and plausibility of its dominance by the following function:

Definition 32: *The salience of attribute classes is measured by the function* $SL : \bigcup_{i=1}^{\infty} A_i \rightarrow [0,...,1]$, *which is defined as*

$$SL(j) = \frac{Bel(DOM_j) + P^*(DOM_j)}{2} \quad \forall \; j \in \bigcup_{i=1}^{\infty} A_i$$

2.3.4 An Example of Salience Estimates

In the following, we give an example of salience estimates in reference to the conceptual schema of figure 9, that is a generalization taxonomy of classes of employees in a research institute. The numbers in the right lowers of thick boxes indicate the cardinalities of the extensions of the relevant classes.

Table 3 presents belief and plausibility measures to the charactericity, abstractness, determinance and dominance of attributes as well as their salience measures(subscripts 1 and 2 distinguish between measures obtained considering all kinds of dependencies or only total equivalencies, respectively).

As shown in this table, the attribute classes *employmentAggreement* and *qualification* are the more salient ones. They are not only abstract(cf. column m_i^a in table 3) but also they are specialized by most of the subclasses of employees (thus becoming characteristic of the relevant generalization taxonomy). Furthermore, a dependency mapping between them cannot be precluded(cf. columns $Bel_1(\overline{DET_i})$ and $Bel_2(\overline{DET_i})$ in table 3) as a result of non common scope or significant numbers of non common refinements. In fact, the qualifications of an employee determine the kind of the job he/she is eligible for and thus the type of the contract he/she may have.

At the other end of the spectrum, the attribute class *keepsAccountsOf* is neither abstract nor determinant (cf. columns m_i^a, $Bel_1(\overline{DET_i})$ and $Bel_2(\overline{DET_i})$ in table 3). Also, it is characteristic only for the

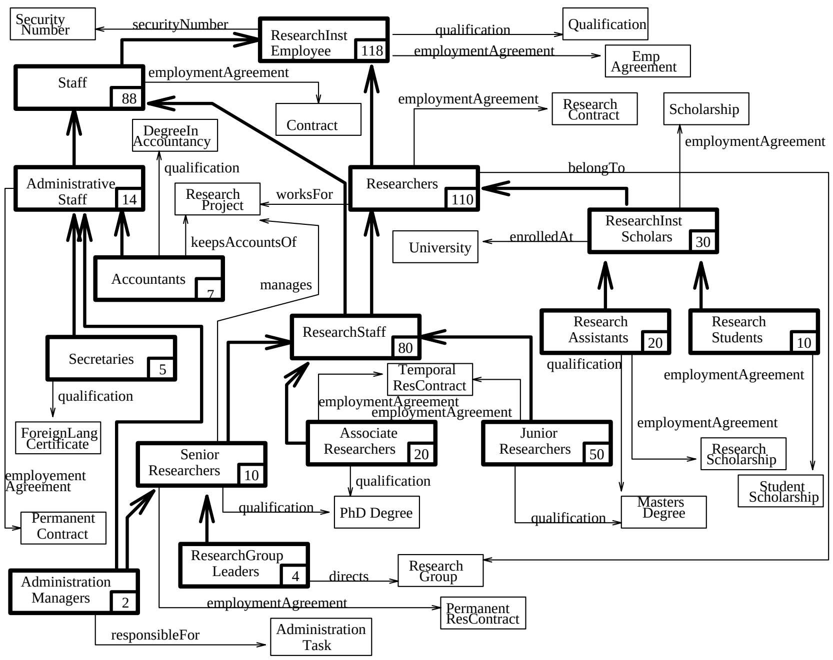

Figure 9: An Example for Estimating Salience of Attributes

Table 3: Evidence and Salience Measures for Attributes in Figure 9

Attribute	m_i^{ch}	m_i^a	$Bel_1(\overline{DET_i})$	$Bel_2(\overline{DET_i})$	$Bel(DOM_i)$	$P*_1(DOM_i)$	$P*_2(DOM_i)$	$SL(i)$
employmentAggreement	0.53	1	0	0.17	0.53	1	0.92	0.765
qualification	0.4	1	0	0.15	0.4	1.0	0.91	0.7
enrolledAt	0.33	1	0.25	0.47	0.33	0.83	0.68	0.58
manages	0.33	0.6	0.1	0.28	0.198	0.92	.77	0.55
worksFor	0.1	1	0	0	0.1	1.0	1.0	0.55
belongsTo	0.1	1	0	0	0.1	1.0	1.0	0.55
securityNumber	0.06	1	0	0	0.06	1.0	1.0	0.503
responsibleFor	1	0	0.44	0.64	0	0.56	0.36	0.28
directs	1	0	0.44	0.64	0	0.56	0.36	0.28
keepsAccountsOf	1	0	0.81	0.77	0	0.19	0.23	0.095

particular kind of employees it applies to(i.e. *Accountants*) but not for the entire taxonomy.

3. Computation of Distance and Salience Functions- Analysis of Complexity

i) The Computation of the Classification Distance

The computation of the classification distance between two objects retrieves their classes, identifies the non-common of them and estimates their specialization depths. Its worst-case time complexity is $O(C_1^3 + C_2^3)$, where C_1 and C_2 are the numbers of the classes of the involved objects, since it might be necessary to estimate the specialization depths for each of them if the involved objects have no classes in common. This step requires the retrieval of the transitive closures of their own superclasses, which has a complexity $O(C_1^2 + C_2^2)$ (the complexity of spanning the transitive closure of a graph with N nodes is N^2 [50]).

ii) The Computation of the Generalization Distance

The computation of the generalization distance between two individual classes retrieves their superclasses, identifies the non-common of them and estimates their specialization depths. In the case of attribute classes, it retrieves their original classes and checks their identity. Similarly to the computation of the classification distance, the computation of the generalization distance has a worse-case time complexity of $O(S_1^3 + S_2^3)$, where S_1 and S_2 are the numbers of the superclasses of the involved objects.

iii) The Computation of the Attribution Distance

To compute the attribution distance between two objects, initially their semantically homogeneous attributes are identified and then their saliences as well as the pairwise distances between them are estimated. The complexity of testing the semantic homogeneity of two attributes x and y is $O(N_x^2 + N_x \log N_x + \sum_{z \varepsilon Classes(x)} N_z \log N_z + N_y^2 + N_y \log N_y + \sum_{w \varepsilon Classes(y)} N_w \log N_w)$ where N_x , N_y , N_z and N_w are the numbers of the classes of the attributes x,y,z and w, as proved in [61].

Then, the selection of the minimum distance isomorphism between the semantically homogeneous attributes of two objects is a *weigthed bipartite graph matching problem*[50]:

Theorem 13: *The selection of the minimum distance isomorphism in computing function D_4 is a weighted bipartite graph matching problem.*

Thus, the minimum distance isomorphism between the attributes of two objects can be selected by an algorithm solving this problem. A traditional such algorithm is the *Hungarian Method*[50], whose complexity is $O(n^3)$ for graphs with *n* nodes in total. Therefore, the time complexity of selecting the minimum distance isomorphism between the attributes of two objects x and y is $O(A_x^3 + A_y^3)$, where A_x and A_y are the numbers of their attributes[61].

iv) The Computation of Salience

The computation of an attribute's salience is based on the estimation of the beliefs and plausibilities to the determinance of the original classes of the attribute classes it has been an instance of (cf. definitions 19 and 28). To obtain these measures, it is necessary to retrieve the *scope*, the *refining classes*, the *possible ranges* and the *original domain class* (together with its *extension*) for each of these original classes and use them in estimating the beliefs to the charactericity and the abstractness of the involved attribute. Furthermore, to estimate the belief to its *non-determinance*, it is necessary to retrieve the original classes of all the other attributes of the classes in its scope (together with their own refining and possible range classes). As discussed in[61], the worse-case time complexity of this computation is $O(L(NM+M^2\log M))$, where L is the number of the attributes of the classes in the scope of the attribute in hand, N is the number of the instances of its original domain class and M is the number of the classes in its scope.

v) The Computation of the Overall Distance

The computation of the overall distance between objects aggregates their partial distances. Due to the recursive computation of the attribution distance, the time complexity of the computation of the overall distance is given by the following fomula(x denotes the level of recursion):

$$T_x = K_{in} + K_{isa} + (A_1 + A_2)K_{sal} + (A_1 A_2)K_{map} + (A_1^3 + A_2^3) + (\sum_i x_i y_i + 1)T_{x-1} \quad (1)$$

where

- K_{in} is the cost of estimating the classification distance of two objects(see relevant complexity above)
- K_{isa} is the cost of estimating the generalization distance of two objects(see relevant complexity above)
- K_{map} is the cost of testing the semantic homogeneity of two attributes of the involved objects(see relevant complexity above)
- K_{sal} is the cost of estimating the salience of an attribute(see relevant complexity above)
- x_i , y_i are the numbers of attributes of the involved objects in each category of semantic homogeneity i (thus, $\sum_i x_i = A_1$, $\sum_i y_i = A_2$).

Formula (1) can be rewritten as:

$$T_x = AT_{x-1} + B \quad (2)$$

where $A = \sum_i x_i y_i + 1$ and $B = K_{in} + K_{isa} + (A_1 + A_2)K_{sal} + (A_1 A_2)K_{map} + (A_1^3 + A_2^3)$

Assuming that, the factors A and B are constant at the different levels of recursion x, formula 2 is a *linear first-order difference equation* and its general solution is $T_x = B\frac{A^x - 1}{A - 1} + CA^x$ where $C = T_0$ [68]. T_0 is the cost of similarity analysis between objects having no attributes and equals to $K_{in} + K_{isa}$ since the computation of the similarity between such objects depends solely on their classification and generalization distances.

Thus, controlling recursion while computing the attribution distance makes the complexity of similarity analysis polynomial. For instance, the termination of recursion after having examined the attributes at the first two levels of the attribution graphs of two objects results in a complexity $B(A+1) + CA^2$. Under controlled recursion, the selected isomorphism between attributes will only approximate the optimal one with a quality depending on the correlation between the classification, generalization and the attribution distances between the relevant objects. However, to the extent that our experiments have shown positive and statistically significant such correlations (cf. table 1 in section 2.2.5), we can expect good such approximations.

4. Empirical Evaluation of the Model

The prescribed similarity model has been implemented in C++[62] and integrated with the *Semantic Index System(SIS)* (see [11, 74] for details). The SIS is a tool for representing scientific knowledge and engineering designs. The resulted prototype was used in preliminary experiments concerning: (1) the consistency of estimates produced by our model with similarity assessments provided by humans(consistency is a prerequisite for applying the model to tasks, where analogical reasoning has to be carried out in cooperation with humans); and (2) the recall performance of similarity analysis in retrieval.

4.1 Ordering Consistency

In the first experiment, we used a conceptual model of the C++ language[62] representing concepts of C++(e.g classes, operators, member functions), the normal file structure of C++ programs(e.g dependencies between source code files and header files) and dependencies between C++ program elements (e.g. function calls, access of variables). The model was developed to support the static analysis of C++ programs[15] and its accuracy regarding the representation of C++ concepts has been evidenced through this application.

Table 4: Rank Correlations Between Model and Human Similarity Orderings

Case	Set Size	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
1	17	0.37	0.5 *	0.47 *	0.37	0.46 *
2	37	0.84 *	0.75 *	0.57 *	0.60 *	-0.06
3	18	-0.32	-	0.5 *	-0.13	0.12
4	18	-0.12	-	0.5 *	-0.18	0.10
5	4	0.55	0.55	0.55	0.55	0.55
6	17	0.55 *	0.60 *	0.55 *	0.51 *	0.64 *
7	4	0.55	0.85	0.65	0.55	0.4
8	18	0.57 *	0.48 *	0.87 *	0.65 *	0.52 *
9	17	0.42 *	0.71 *	0.39	0.50 *	0.5 *
10	19	0.52 *	0.87 *	0.73 *	0.86 *	0.64 *

Ten cases of similarity analysis were carried out. In each case, the elements of a source set of C++ concepts were ranked in descending similarity order with respect to a target C++ concept. The similarities between the compared concepts in each of these cases were measured from their conceptual descriptions in the employed model of C++. The same cases were given to five software engineers, ignorant of the similarity analysis imposed ordering, who were asked to rank the source concepts with respect to their similarities to the target(as they perceived them). The subjects had different degrees of expertise and knowledge of the C++ conceptual model(decreasing along with increasing subject numbers in table 4).

The *rank correlations* of the two orderings were measured according to the Spearman coefficient[25,33]. As shown in table 4, a positive rank correlation was found in all but 5 cases. Most of the positive coefficients(28 out of 44) were found statistically significant(i.e. coefficients marked by an asterisk) according to the *D or t criteria*[33] at the level $p < 0.05$, unlike the negative ones. These preliminary findings indicate that the similarity model generates measures, which do not violate human intuition about similarity.

4.2 Recall Evaluation

Recall measures, as indicators of completeness in retrieval[63], associate with the possibility of ignoring relevant analogs in an analogical reasoning session after analogical retrieval (high recall indicates a low such possibility and vice versa).

Using exactly the same C++ conceptual model and four of the subjects participated in the first experiment we also measured the *recall* performance of the similarity model. The subjects were given the same 10 cases of the previous experiment and were asked to indicate whether each of the source C++ concepts could be used instead of the target one in some programming task they could think of. Their answers - interpreted as indicating the analogy between the relevant concepts in programming - were used as judgements of relevance[63]. On the basis of these judgements, we measured the recall performance of the model in reference to the source sets. Recall was measured according to the following formula:

$$r = a_c / R$$

where a_c was the number of the relevant sources in the first c % positions of the sorted(in descending similarity order) source set and R the total number of the relevant items in it[32].

Table 5 presents measured recalls for different subjects and cutoff levels. In our view, these measures are encouraging since the first two of the selected cutoffs were fairly low and the proportion of the items in any of the source sets, judged as relevant by any of the subjects did not exceed 25%. In fact, only 10% of items in the source sets were assessed as being relevant on the average.

Table 5: Average Recall for Different Cutoffs					
Cutoff	Subject 1	Subject 2	Subject 3	Subject 5	General
10%	0.33	0.35	0.34	0.44	0.35
25%	0.54	0.40	0.74	1.00	0.60
50%	0.83	0.80	0.91	1.00	0.86

5. Related Work

This section assesses the merits of the similarity and other computational models of analogical reasoning with respect to criteria related to pragmatic aspects of their application. These criteria include: (1) the ability of a model to elaborate analogies between objects in different domains(i.e. domain independence); (2) the need of causal knowledge and/or user-supplied information for the elaboration of analogies and; (3) the computational complexity of a model. The former two criteria shape the spectrum of the tasks a model might be applicable to, while the latter determines the efficiency of its application.

Computational models of analogical reasoning can be distinguished into:

(1) *general* models which detect analogies without distinguishing the purpose of reasoning and the types of the involved analogs(i.e. the *Structure Mapping Engine(SME)*[17], the *Constraint Satisfaction Theory(CST)* implemented by ACME [28] and ARCS[65] and the *Constrained Semantic Transference(CST)*[29]);

(2) models of *analogical understanding* which detect analogies to explain unfamiliar domains in terms of familiar ones, (e.g. MACKBETH[72] and CARL[7]);

(3) models of *analogical problem solving*, which detect analogies to solve specific problems and involve analogs from different domains(e.g. ANA[41], explicit planning[45], MEDIATOR[37], ARCHES[5], purpose-directed analogy[30], transformational and derivational analogy[8, 9, 43], NLANG[23], PRODIGY[69] and ANAGRAM[13]; and,

(4) models of *case-based reasoning*, a special kind of models of analogical problem solving, which detect analogies between items in the same application domain(e.g. PARADYME/JULIA[36], SURVER III[73], HYPO[4], CADET[59], CABOT[12], Protos[48], FGP[16], CBL4[2] and CBR+EBL[14]).

In general, the models of the latter three types utilize causal knowledge determining the elements of analogs, which are important for analogies sought for a particular purpose. This knowledge usually enables a fast detection of analogies, by limiting the search space of the analog-elements that must be taken into account. However, there might be domains which totally or partially lack such knowledge or where it is very expensive to acquire it. Also, the substance[61], the granularity[39] and the representation of causal knowledge might differ across domains [39,61]. Hence, the whole approach and consequently the models adopting it may not be always usable.

However, even the general models of analogical reasoning might be weak in the light of the prescribed criteria. In particular:

The SME is domain independent and does not rely on causal knowledge. In fact, the model matches identical relations of analogs, which do not violate the relational structures of their objects according to the *systematicity principle*[20] (i.e. when two relations are mapped their corresponding arguments must be mapped as well). Also the importance of different relations is distinguished syntactically from their arity and order. However, the SME has a combinatorial computational complexity, i.e. $O(N!)$ where N is the number of analog elements that must be considered for mapping.

The CST matches analogs obeying the systematicity principle but it considers only elements, which are known to be important for analogies or related to goal attainment. Also, elements cannot be matched unless they are associated with concepts in a lexicon related by specific types of semantic similarity relations (e.g. *synonymy*, *hyponymy*, *meronymy*[65]). To the extent that these lexicon concepts usually have context and domain dependent meanings[19] and the specific primitives used for expressing causalities between analog elements and goals may not be adequate for all possible domains[61], the model is not domain independent and is sensitive to causal knowledge. Its parallel implementation using connectionist networks does not allow an analytical prediction of its worse-time complexity, although the model is claimed to be efficient[65].

The Constrained Semantic Transference model is domain independent and does not need causal knowledge or explicit user-information for detecting analogies. It maps elements of analogs, which are of the same type. Mappings are constrained by demanding the consistency of the properties of the image of an element with existing knowledge about it(i.e. *coherent mappings*). This formal notion of coherency can only be approximately checked within limited computational time but is not computable in the general case. Hence, the model is of limited practical utility.

In the light of our previous criticism, the similarity model ranks better than other general models when all the prescribed criteria are taken into account. In fact, neither it employs any domain dependent matching criteria, nor it relies on special types of causal knowledge or user-supplied information for detecting analogies. Also, as discussed in section 3, it has a polynomial complexity.

6. Conclusions and Future Research

In this paper, we defined and analysed a computational model for elaborating analogies between conceptual objects. The model consists of metric functions, which measure the conceptual distances between objects with respect to their classifications, generalizations and attributes as well as functions which measure the importance of these semantic modelling elements to the existence of analogies. Its competence in detecting analogies depends on how precisely the semantics of objects are expressed by their conceptual descriptions. Improper conceptual modelling deteriorates this competence and cannot be identified by similarity analysis, itself.

Because of the formal definition of its distance and salience measuring functions the model is amenable to analytic assessment. Also, it is domain independent and operational upon non uniform representations of objects. It does not need causal knowledge or explicit, user-supplied information for detecting analogies and has a polynomial complexity. These characteristics give it a potential of being applicable to complex tasks involving intra or inter-domain analogical reasoning, some of which have been investigated elsewhere[52,53,54,75,76,77].

A further and extensive empirical evaluation of the model is necessary for validating the initial empirical findings presented in this paper and the positive evidence acquired from applying it to tasks of software reuse[52,53,54,55]. Such an evaluation is being planned within real application environments and tasks varying with respect to the goals they impose and the conceptual models they use for describing potential analogs. The implementation of the model upon the Semantic Index System platform(see [11,74] for details) makes this application feasible regarding information management and usability requirements[11].

We are also designing a parallel implementation of the model. In fact, the classification, generalization and attribution distances can be computed in parallel because they do not depend on each other. Also, since the semantic homogeneity is an equivalence relation, it partitions the attributes of objects into disjoint categories. This makes possible the parallel computation of partial optimal isomorphisms for each such category followed by their merging into an overall optimal isomorphism, while computing the attribution distance between objects(cf. proof of theorem 13 in Appendix 2).

Appendix 1: Axiomatic Foundation of Metric Functions and Dempster-Shafer Beliefs

i) Metrics

Following [21], a function *d* on a non empty set X is a metric for it, if it assigns to each pair of its elements of a real number(i.e. $d : X \times X \to R$), which satisfies the following axioms:

(α_1) $D(x,y) \geq 0$

(α_2) $D(x,y) \equiv D(y,x)$

(α_3) $D(x,z) \leq D(x,y) + D(y,z)$

(α_4) $D(x,y) = 0 \Longleftrightarrow x = y$

A function *d* which satisfies only the first three of these axioms is called *pseudometric*[34].

ii) Elements of the Dempster-Shafer Theory of Evidence

Belief in the context of the Dempster-Shafer theory of evidence[57] is a function

$Bel : PowersetOf(\Theta) \to [0,...1]$

obeying the axioms:

(α_5) $Bel(P) = 0$, if $P = \varnothing$

(α_6) $Bel(\Theta) = 1$

(α_7) $Bel(\bigcup_{i=1}^{n} P_i) \geq \sum_{I \subseteq \{1,2,...,n\}, I \neq \varnothing} (-1)^{|I|+1} Bel(\bigcap_{i \, \varepsilon \, I} P_i)$

where Θ is a set of mutually exclusive propositions referred to as the *frame of discernment*, $n = |PowersetOf(\Theta)|$ and P, P_i are subsets of Θ denoting the logical disjunction of their elements.

The total belief committed to a subset P, i.e. $Bel(P)$, results from accumulating *basic probability assignments* to its subsets:

(f 1) $Bel(P) = \sum_{A \subseteq P} m(A)$

A basic probability assignment is a function *m* obeying the following axioms:

(α_8) $m : PowersetOf(\Theta) \to [0,...,1]$

(α_9) $m(\varnothing) = 0$

(α_{10}) $\sum_{P \subseteq \Theta} m(P) = 1$

Different basic probability assignments arised from distinct bodies of evidence can be combined according to the formula:

$$m_1 \bar{\underline{+}}\ m_2(P) = \frac{\sum_{X \cap Y = P} m_1(X) m_2(Y)}{1 - k_0} \text{ where } k_0 = \sum_{Z \cap W = \varnothing,\, Z \subseteq \Theta,\, W \subseteq \Theta} m_1(Z) m_2(W)$$

which is known as the rule of the *orthogonal sum* [57].

Appendix 2: Proofs of Theorems in Sections 2 and 3

Lemma 1: *For any three sets A,B,D it holds that*

$(A-B)\bigcup(B-A)\subseteq((A-D)\bigcup(D-A))\bigcup((D-B)\bigcup(B-D))$.

Proof: It can be proved by algebraic manipulation as in [61]. □

Lemma 2: *The homographic transformation* $d(x_1,x_2)=\dfrac{aD(x_1,x_2)}{aD(x_1,x_2)+1}$, $a>0$ *of a metric D is a metric*

Proof: It can be proved by algebraic manipulation as in [61]. □

Theorem 1: *Function* d_2 is a pseudometric.

Proof: Function D_2 is pseudometric since:

(i) $D_2(\#i,\#j)=\sum_{x\ \varepsilon\ (o(\#i).In-o(\#j).In)}\dfrac{1}{SD(x)}+\sum_{x\ \varepsilon\ (o(\#j).In-o(\#i).In)}\dfrac{1}{SD(x)}\geq 0$ (i.e. axiom α_1))

(ii) $D_2(\#i,\#j)=\sum_{x_1\ \varepsilon\ S_{ij}}\dfrac{1}{SD(x_1)}+\sum_{x_2\ \varepsilon\ S_{ji}}\dfrac{1}{SD(x_2)}=\sum_{x_2\ \varepsilon\ S_{ji}}\dfrac{1}{SD(x_2)}+\sum_{x_1\ \varepsilon\ S_{ij}}\dfrac{1}{SD(x_1)}=D_2(\#j,\#i)$ (if $S_{ij}=\left\{o(\#i).In\bigcup-o(\#j).In\right\}$ and $S_{ji}=\left\{o(\#j).In-o(\#i).In\right\}$) (i.e. axiom α_2)) and

(iii) $\sum_{x\ \varepsilon\ S_1}\dfrac{1}{SD(x)}\leq\sum_{x\ \varepsilon\ (S_2\bigcup S_3)}\dfrac{1}{SD(x)}\leq\sum_{x\ \varepsilon\ S_2}\dfrac{1}{SD(x)}+\sum_{x\ \varepsilon\ S_3}\dfrac{1}{SD(x)}$ if $S_1=(S_i-S_j)\bigcup(S_j-S_i)$ $S_2=(S_i-S_l)\bigcup(S_l-S_i)$ $S_3=(S_l-S_j)\bigcup(S_j-S_l)$ ($S_i=o(\#i).In$, $S_j=o(\#j).In$, $S_l=o(\#l).In$) since by *lemma 1* $S_1\subseteq S_2\bigcup S_3$ (i.e. axiom α_3)) . Then by *lemma 2* d_2 is also a pseudometric. □

Theorem 2: *Function* d_3 is a metric .

Proof: It can be proved in a way similar to theorem 1 as in [61]. □

Theorem 3: *The semantic homogeneity of attributes is an equivalence relation*

Proof: The semantic homogeneity is a reflexive (i.e. $sh(\#i,\#i)$) since $OCL[\#i]\equiv OCL[\#i]$, symmetric (i.e. $sh(\#i,\#j)\Longleftrightarrow sh(\#j,\#i)$) since $OCL[\#i]=OCL[\#j]\Rightarrow OCL[\#j]=OCL[\#i]$ and transitive relation (i.e. $sh(\#i,\#j)$ and $sh(\#j,\#l)\Rightarrow sh(\#i,\#l)$) since $(OCL[\#i]=OCL[\#j])$ and $(OCL[\#j]=OCL[\#l])\Rightarrow(OCL[\#i]=OCL[\#l])$ □

Lemma 3: *Function* D_4 is a metric.

Proof: a) D_4 satisfies axiom α_1 when $INT[\#i]=\varnothing$ *or* $INT[\#j]=\varnothing$ since $D_4(\#i,\#j)=\infty$. Also, when both #i and #j have attributes, let us assume that one attribute of each element (x_1,x_2) of their optimal isomorphism c_k has no attributes of its own. This is a valid assumption since eventually some of the $INT[x_1]$ or $INT[x_2]$ will be empty(even due to $V=IU\bigcup AU$). In this case, $D_4(x_1,x_2,V)\geq 0$, as it was just proved, and by *lemmas 4* and *5* $D'(x_1,x_2,V)\geq 0$ *and* $d'(x_1,x_2,V)\geq 0$, respectively. Therefore,

$\min_{c_k \, \varepsilon \, C[\#i,\#j]} \left\{ \sum_{(x_1,x_2)\, \varepsilon \, c_k} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_3 \, \varepsilon \, A_{\#i}[c_k]} s(x_3)^2 + \sum_{x_4 \, \varepsilon \, A_{\#j}[c_k]} s(x_4)^2 \right\} \geq 0$ or equivalently $D_4(\#i,\#j,V) \geq 0$.

b) D_4 is symmetric by definition when $INT[\#i] = \varnothing$ *or* $INT[\#j] = \varnothing$. When, #i or #j have their own attributes, each of the isomorphisms c_k in C[#i,#j] will have an inverse isomorphism c'_k ($c'_k = c_k^{-1}$) in C[#j,#i] and furthermore $A_{\#i}[c'_k] = A_{\#i}[c_k]$ and $A_{\#j}[c'_k] = A_{\#j}[c_k]$. Assuming that at least one attribute of each element (x_1,x_2) in combinations c_k and c'_k has no attributes of its own, $D_4(x_1,x_2,V)$=$D_4(x_2,x_1,V)$ (as already proved) and therefore $d'(x_1,x_2,V) = d'(x_2,x_1,V)$ (by *lemma 5*). Thus, forall c_k , c'_k :

$\sum_{(x_1,x_2)\, \varepsilon \, c_k} s(x_1)s(x_2)d'(x_1,x_2,V) = \sum_{(x_2,x_1)\, \varepsilon \, c'_k} s(x_2)s(x_1)d'(x_2,x_1,V)$ and

$\sum_{x_3 \, \varepsilon \, A_{\#i}[c_k]} s(x_3)^2 = \sum_{x_3 \, \varepsilon \, A_{\#i}[c'_k]} s(x_3)^2$ and $\sum_{x_4 \, \varepsilon \, A_{\#j}[c_k]} s(x_3)^2 = \sum_{x_4 \, \varepsilon \, A_{\#j}[c'_k]} s(x_3)^2$

Therefore,

$$\min_{c_k \, \varepsilon \, C[\#i,\#j]} \left\{ \sum_{(x_1,x_2)\, \varepsilon \, c_k} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_3 \, \varepsilon \, A_{\#i}[c_k]} s(x_3)^2 + \sum_{x_4 \, \varepsilon \, A_{\#j}[c_k]} s(x_4)^2 \right\} =$$
$$\min_{c'_k \, \varepsilon \, C[\#j,\#i]} \left\{ \sum_{(x_2,x_1)\, \varepsilon \, c'_k} s(x_2)s(x_1)d'(x_2,x_1,V) + \sum_{x_3 \, \varepsilon \, A_{\#i}[c'_k]} s(x_3)^2 + \sum_{x_4 \, \varepsilon \, A_{\#j}[c'_k]} s(x_4)^2 \right\}$$

or equivalently $D_4(\#i,\#j,V) = D_4(\#j,\#i,V)$.

c) D_4 is triangular by definition when one, two or all of three objects #i,#j and #l have no attributes When all #i , #j and #l have attributes, let C_{il}^{opt} and C_{lj}^{opt} be the optimal isomorphisms between the attributes of the objects #i, #l and #l, #j, respectively and, C'_{ij} be an isomorphism between the attributes of #i and #j defined as(see transitivity of semantic homogeneity):

$$C'_{ij} = \left\{ (x_1,x_3) \mid (\exists\, x_2 : ((x_1,x_2)\ \varepsilon\ C_{il}^{opt}) \text{ and } ((x_2,x_3)\ \varepsilon\ C_{lj}^{opt})) \right\} \qquad \text{(I)}$$

Given C'_{ij} , it is sufficient to prove that

$$\sum_{(x_1,x_2)\, \varepsilon \, C'_{ij}} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_3 \, \varepsilon \, A_{\#i}[C'_{ij}]} s(x_3)^2 + \sum_{x_4 \, \varepsilon \, A_{\#j}[C'_{ij}]} s(x_4)^2 \leq$$
$$\sum_{(x_5,x_6)\, \varepsilon \, C_{il}^{opt}} s(x_5)s(x_6)d'(x_5,x_6,V) + \sum_{x_7 \, \varepsilon \, A_{\#i}[C_{il}^{opt}]} s(x_7)^2 + \sum_{x_8 \, \varepsilon \, A_{\#l}[C_{il}^{opt}]} s(x_8)^2 +$$
$$\sum_{(x_9,x_{10})\, \varepsilon \, C_{lj}^{opt}} s(x_9)s(x_{10})d'(x_9,x_{10},V) + \sum_{x_{11} \, \varepsilon \, A_{\#l}[C_{lj}^{opt}]} s(x_{11})^2 + \sum_{x_{12} \, \varepsilon \, A_{\#j}[C_{lj}^{opt}]} s(x_{12})^2 \qquad \text{(II)}$$

because even if C'_{ij} is not the optimal isomorphism C_{ij}^{opt} , between the objects #i , #j, C_{ij}^{opt} will result into a less total distance according to definition 19. However, if:

$$C'_{il} = \left\{ (x_1,x_2) \mid ((x_1,x_2)\ \varepsilon\ C_{il}^{opt}) \text{ and } (\nexists\, x_3 : ((x_1,x_3)\ \varepsilon\ C'_{ij}) \right\} \qquad \text{(III)}$$

and

$$C'_{lj} = \left\{ (x_1,x_2) \,|\, ((x_1,x_2) \ \varepsilon\ C_{lj}^{opt}) \text{ and } (\exists\ x_3 : ((x_3,x_2) \ \varepsilon\ C'_{ij}) \right\} \qquad \text{(IV)}$$

then,

$$\sum_{(x_1,x_2)\ \varepsilon\ C_{il}^{opt}} s(x_1)s(x_2)d'(x_1,x_2,V) =$$

$$\sum_{(x_3,x_4)\ \varepsilon\ C'_{il}} s(x_3)s(x_4)d'(x_3,x_4,V) + \sum_{(x_5,x_6)\ \varepsilon\ (C_{il}^{opt}-C'_{il})} s(x_5)s(x_6)d'(x_5,x_6,V)$$

and

$$\sum_{(x_1,x_2)\ \varepsilon\ C_{lj}^{opt}} s(x_1)s(x_2)d'(x_1,x_2,V) =$$

$$\sum_{(x_3,x_4)\ \varepsilon\ C'_{lj}} s(x_3)s(x_4)d'(x_3,x_4,V) + \sum_{(x_5,x_6)\ \varepsilon\ (C_{lj}^{opt}-C'_{lj})} s(x_5)s(x_6)d'(x_5,x_6,V)$$

Also, if we define:

$$A_{\#i}^{(1)}[C'_{ij}] = \left\{ x_1 \,|\, (not(\exists\ x_2 : (x_1,x_2) \ \varepsilon\ C_{il}^{opt} \right\}$$

$$A_{\#i}^{(2)}[C'_{ij}] = \left\{ x_1 \,|\, (\exists\ x_2 : ((x_1,x_2) \ \varepsilon\ C_{il}^{opt})) \text{ and } (not(\ \exists\ x_3 : ((x_2,x_3) \ \varepsilon\ C_{lj}^{opt}))) \right\} \text{ and}$$

$$A_{\#j}^{(1)}[C'_{ij}] = \left\{ x_2 \,|\, (not(\exists\ x_1 : (x_1,x_2) \ \varepsilon\ C_{lj}^{opt} \right\}$$

$$A_{\#j}^{(2)}[C'_{ij}] = \left\{ x_2 \,|\, (\exists\ x_1 : ((x_1,x_2) \ \varepsilon\ C_{lj}^{opt})) \text{ and } (not(\ \exists\ x_3 : ((x_3,x_1) \ \varepsilon\ C_{il}^{opt}))) \right\}$$

then

$$A_{\#i}[C'_{ij}] = A_{\#i}^{(1)}[C'_{ij}] \cup A_{\#i}^{(2)}[C'_{ij}] \ \text{ and } \ A_{\#i}^{(1)}[C'_{ij}] = A_{\#i}[C_{il}^{opt}]$$

$$A_{\#j}[C'_{ij}] = A_{\#j}^{(1)}[C'_{ij}] \cup A_{\#j}^{(2)}[C'_{ij}] \ \text{ and } \ A_{\#j}^{(1)}[C'_{ij}] = A_{\#j}[C_{lj}^{opt}]$$

Thus, (II) becomes,

$$\sum_{(x_1,x_2)\ \varepsilon\ C'_{ij}} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_3\ \varepsilon\ A_{\#i}[C_{il}^{opt}]} s(x_3)^2 + \sum_{x_4\ \varepsilon\ A_{\#i}^{(2)}[C'_{ij}]} s(x_4)^2 + \sum_{x_5\ \varepsilon\ A_{\#j}[C_{lj}^{opt}]} s(x_5)^2 + \sum_{x_6\ \varepsilon\ A_{\#j}^{(2)}[C'_{ij}]} s(x_6)^2 \leq$$

$$\sum_{(x_7,x_8)\ \varepsilon\ C'_{il}} s(x_7)s(x_8)d'(x_7,x_8,V) + \sum_{(x_9,x_{10})\ \varepsilon\ (C_{il}^{opt}-C'_{il})} s(x_9)s(x_{10})d'(x_9,x_{10},V) + \sum_{x_{11}\ \varepsilon\ A_{\#i}[C_{il}^{opt}]} s(x_{11})^2 + \sum_{x_{12}\ \varepsilon\ A_{\#l}[C_{il}^{opt}]} s(x_{12})^2 +$$

$$\sum_{(x_{13},x_{14})\ \varepsilon\ C'_{il}} s(x_{13})s(x_{14})d'(x_{13},x_{14},V) + \sum_{(x_{15},x_{16})\ \varepsilon\ (C_{lj}^{opt}-C'_{lj})} s(x_{15})s(x_{16})d'(x_{15},x_{16},V) + \sum_{x_{17}\ \varepsilon\ A_{\#j}[C_{lj}^{opt}]} s(x_{17})^2 + \sum_{x_{18}\ \varepsilon\ A_{\#l}[C_{lj}^{opt}]} s(x_{18})^2 \qquad \text{(V)}$$

and equivalently,

$$\sum_{(x_1,x_2)\,\varepsilon\, C'_{ij}} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_4\,\varepsilon\, A^{(2)}_{\#i}[C'_{ij}]} s(x_4)^2 + \sum_{x_6\,\varepsilon\, A^{(2)}_{\#j}[C'_{ij}]} s(x_6)^2 \leq$$

$$\sum_{(x_7,x_8)\,\varepsilon\, C'_{il}} s(x_7)s(x_8)d'(x_7,x_8,V) + \sum_{(x_9,x_{10})\,\varepsilon\,(C^{opt}_{il}-C'_{il})} s(x_9)s(x_{10})d'(x_9,x_{10},V) + \sum_{x_{12}\,\varepsilon\, A_{\#l}[C^{opt}_{il}]} s(x_{12})^2 +$$

$$\sum_{(x_{13},x_{14})\,\varepsilon\, C'_{il}} s(x_{13})s(x_{14})d'(x_{13},x_{14},V) + \sum_{(x_{15},x_{16})\,\varepsilon\,(C^{opt}_{lj}-C'_{lj})} s(x_{15})s(x_{16})d'(x_{15},x_{16},V) + \sum_{x_{18}\,\varepsilon\, A_{\#l}[C^{opt}_{lj}]} s(x_{18})^2 \qquad \text{(VI)}$$

However, according to the definition of the isomorphisms C'_{ij} , C'_{il} , C'_{lj} , we have that:

$$\forall\ x_1, x_3 : ((x_1,x_3)\ \varepsilon\ C'_{ij}) \Rightarrow (\ \exists\ x_2 : ((x_1,x_2)\ \varepsilon\ C'_{il}) \text{ and } ((x_2,x_3)\ \varepsilon\ C'_{lj})) \text{ and}$$

$$sh(x_1,x_3) \text{ and } sh(x_1,x_2) \text{ and } sh(x_2,x_3)$$

Thus, x_1 ,x_2 and x_3 must belong to the same original attribute classes and by definition 19:

$$s(x_1) = s(x_2) = s(x_3) \qquad \text{(VII)}$$

Let us also assume that at least one attribute x of each element in C'_{ij} , C'_{il} , C^{opt}_{il} , C'_{lj} , C^{opt}_{lj} has no attributes of its own, as in parts (a) and (b). In this case, D_4 has been already shown to be triangular. Therefore by *lemma 4* D' is triangular and by *lemma 5*:

$$d'(x_1, x_3,V) \leq d'(x_1, x_2,V) + d'(x_2, x_3,V) \qquad \text{(IIX)}$$

From (VII),(IIX) we have that:

$$\sum_{(x_1,x_2)\,\varepsilon\, C'_{ij}} s(x_1)s(x_2)d'(x_1,x_2,V) \leq \sum_{(x_1,x_3)\,\varepsilon\, C'_{il}} s(x_1)s(x_3)d'(x_1,x_3,V) + \sum_{(x_2,x_3)\,\varepsilon\, C'_{lj}} s(x_2)s(x_3)d'(x_2,x_3,V) \Longleftrightarrow$$

$$\sum_{(x_1,x_2)\,\varepsilon\, C'_{ij}} s(x_1)^2 d'(x_1,x_2,V) \leq \sum_{(x_1,x_3)\,\varepsilon\, C'_{il}} s(x_1)^2 d'(x_1,x_3,V) + \sum_{(x_2,x_3)\,\varepsilon\, C'_{lj}} s(x_1)^2 d'(x_2,x_3,V)$$

Thus, for proving (VI) it is sufficient to prove that:

$$\sum_{x_4\,\varepsilon\, A^{(2)}_{\#i}[C'_{ij}]} s(x_4)^2 + \sum_{x_6\,\varepsilon\, A^{(2)}_{\#j}[C'_{ij}]} s(x_6)^2 \leq$$

$$\sum_{(x_9,x_{10})\,\varepsilon\,(C^{opt}_{il}-C'_{il})} s(x_9)s(x_{10})d'(x_9,x_{10},V) + \sum_{x_{12}\,\varepsilon\, A_{\#l}[C^{opt}_{il}]} s(x_{12})^2 +$$

$$\sum_{(x_{15},x_{16})\,\varepsilon\,(C^{opt}_{lj}-C'_{lj})} s(x_{15})s(x_{16})d'(x_{15},x_{16},V) + \sum_{x_{18}\,\varepsilon\, A_{\#l}[C^{opt}_{lj}]} s(x_{18})^2 \qquad \text{(IX)}$$

However, due to definitions of $A^{(2)}_{\#i}[C'_{ij}]$, $A^{(2)}_{\#j}[C'_{ij}]$, $A_{\#l}[C^{opt}_{il}]$ and $A_{\#l}[C^{opt}_{lj}]$, we have that:

$$\forall\ x_1 : (x_1\ \varepsilon\ A^{(2)}_{\#i}[C'_{ij}]) \Rightarrow (\ \exists\ x_2 : ((x_1,x_2)\ \varepsilon\ C^{opt}_{il}) \text{ and } (not(\ \exists\ x_3 : ((x_2,x_3)\ \varepsilon\ C^{opt}_{lj}))))$$

or, equivalently, $\forall\ x_1 : (x_1\ \varepsilon\ A^{(2)}_{\#i}[C'_{ij}]) \Rightarrow (\ \exists\ x_2 : (x_2\ \varepsilon\ A_{\#l}[C^{opt}_{lj}]) \text{ and } sh(x_1, x_2))$

Then according to the restriction of the s(i) factors, it will also be that $s(x_1) = s(x_2)$ and therefore

$$\sum_{x_4 \,\varepsilon\, A^{(2)}_{\#i}[C'_{ij}]} s(x_4)^2 \leq \sum_{x_{18} \,\varepsilon\, A_{\#l}[C^{opt}_{lj}]} s(x_{18})^2 \qquad \text{(X)}$$

Similarly, $\forall\ x_1 : (x_1 \,\varepsilon\, A^{(2)}_{\#j}[C'_{ij}]) \Rightarrow (\ \exists\ x_2 : ((x_2,x_1) \,\varepsilon\, C^{opt}_{lj})$ and $(not(\ \exists\ x_3 : ((x_3,x_2) \,\varepsilon\, C^{opt}_{il})))$

or equivalently, $\forall\ x_1 : (x_1 \,\varepsilon\, A^{(2)}_{\#j}[C'_{ij}]) \Rightarrow (\ \exists\ x_2 : (x_2 \,\varepsilon\, A_{\#l}[C^{opt}_{il}])$ and $sh(x_2, x_1))$

Then according to the restriction of the s(i) factors, it will also be that $s(x_1) = s(x_2)$ and therefore

$$\sum_{x_4 \,\varepsilon\, A^{(2)}_{\#j}[C'_{ij}]} s(x_4)^2 \leq \sum_{x_{18} \,\varepsilon\, A_{\#l}[C^{opt}_{il}]} s(x_{18})^2 \qquad \text{(XI)}$$

(X) and (XI) imply (IX) and equivalently (II). Therefore, D_4 is triangular even if objects have attributes. Thus, (IIX) is true in general and consequently D_4 is triangular in general. □

Theorem 4: *Function* d_4 is a metric.

Proof: This theorem is an implication of *lemmas 3* and *2*. □

Theorem 5: *All the pairs of attribute objects* (x_1,x_2) *of two objects #i, #j that belong to the set SH[#i,#j] and have the same original class belong to the optimal isomorphism* c_{opt} *between the objects #i and #j.*

Proof(Sketch): The axiomatic foundation of the representation framework of similarity analysis, restricts the attributes of an object to have different original classes(see axiom A.3.19 in[61]). Let as also assume that the optimal isomorphism c_{opt} between the attributes of two objects #i and #j does not map the attribute x_1 of #i onto the only attribute x_2 of #j, which has the same original class with it but the attribute x_n. Then if isomorphism c' is defined as $c' = c_{opt} - \left\{(x_1,x_k), (x_n,x_2)\right\} \cup \left\{(x_1,x_2), (x_n, x_k)\right\}$ it can be proved that $d'(x_1,x_2,V) + d'(x_n,x_k,V) < d'(x_1,x_k,V) + d'(x_n,x_2,V)$ or equivalently that

$$\sum_{(x_1,x_2)\,\varepsilon\, c'} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_3 \,\varepsilon\, A_{\#i}[c']} s(x_3)^2 + \sum_{x_4 \,\varepsilon\, A_{\#j}[c']} s(x_4)^2 <$$

$$\sum_{(x_1,x_2)\,\varepsilon\, c_{opt}} s(x_1)s(x_2)d'(x_1,x_2,V) + \sum_{x_3 \,\varepsilon\, A_{\#i}[c_{opt}]} s(x_3)^2 + \sum_{x_4 \,\varepsilon\, A_{\#j}[c_{opt}]} s(x_4)^2$$

which contradicts the initial assumption of c_{opt} being the optimal isomorphism between #i and #j. The proof, as presented in detail in [61], is constructed by algebraic manipulation of this inequality based on $d_3(x_1,x_2) = 0$ and $d_3(x_n,x_2) = d_3(x_1,x_k) = 1$ □

Theorem 6: *Function D is a metric.*

Proof: The metric axioms are proved for the quadric form defined by matric W_{ec} (i.e. $D(x_1,x_2,V) = (d_1(x_1,x_2)^2 + (d_2(x_1,x_2) + d_3(x_1,x_2) + d_4(x_1,x_2,V))^2)^{1/2}$). Similar proofs can be constructed for the other quadric forms, defined by matrices W_{et}, W_{ac} and W_{at}. Axioms α_1 and α_2 are satisfied by

D since they are satisfied by each of the d_1,d_2,d_3 and d_4. Notice also that $D(x_1,x_2)$ is the Euclidean norm of the vector $V_{x_1x_2} = [d_1(x_1,x_2) \quad d_2(x_1,x_2) + d_3(x_1,x_2) + d_4(x_1,x_2,V))]$. Since, the Euclidean norm obeys the triangular inequality(i.e. $||V_{x_1x_2} + V_{x_2x_3}||_2 \le ||V_{x_1x_2}||_2 + ||V_{x_2x_3}||_2$ and $D(x_1,x_3) \le ||V_{x_1x_2} + V_{x_2x_3}||_2$ (because d_1 and $d_2 + d_3 + d_4$ are triangular themeselves) D also obeys axiom α_3. Finally, axiom α_4 is satisfied, since D(#i,#j) cannot be equal to 0 unless $d_1(\#i,\#j)$ equals 0, which is true if and only if #i and #j are equal by definition 12. □

Lemma 4: *Function D' is a metric.*

Proof: It can be obtained exactly as the proof of *theorem 6* replacing D by D'. □

Lemma 5: *Function d' is a metric .*

Proof: This lemma is a consequence of *lemmas 4*, and *2*. □

Theorem 7: *The function* $m_i^{ch}(P)$ *is a basic probability assignment.*

Proof: *Axiom* α_8 : By its definition $m_i^{ch}(P) \ge 0$. Also by definition 11, $\forall x : (x \in AR[\#i]) \rightarrow (\exists y, z : (y \in R[\#i])$ and $(z \in o(y).A)$ and $(o(z).TO = x))$ implying that $AR[\#i] \subseteq R[\#i]$ and $|AR[\#i]| \le |R[\#i]|$. Furthermore, by definition 10, $\forall x : x \in R[\#i] \rightarrow x \in S[\#i]$ and thus $R[\#i] \subseteq S[\#i]$ and $|R[\#i]| \le |S[\#i]|$. Therefore, $e_i = |AR[i]|/|S[\#i]| \le 1$ and $1 - e_i \le 1$ or equivalently, $m_i^{ch} \le 1$.

Axiom α_9 : It is implied by the definition of the focals, CH_i and $\overline{CH_i}$ of m_i^{ch} as non empty sets in the frame of discernment Θ.

Axiom α_{10} : $\sum_{P \subseteq \Theta} m_i^{ch}(P) = m_i^{ch}(CH_i) + m_i^{ch}(\overline{CH_i}) + \sum_{P \subseteq \Theta \,\&\, P \neq CH_i \,\&\, P \neq \overline{CH_i}} m_i^{ch}(P) = c_i + (1 - c_i) + 0 = 1$ □

Theorem 8: *The function* $m_i^{a}(P)$ *is a basic probability assignment.*

Proof: *Axiom* α_8 : By its definition $m_i^{a}(P) \ge 0$. Also, by definition 5 $\forall x : x \in EXT_s[\#j] \rightarrow x \in EXT[\#j]$. Therefore, $EXT_s[\#j] \subseteq EXT[\#j]$ and $|EXT_s[\#j]| \le |EXT[\#j]|$ or equivalently $m_i^{a} \le 1$.

Axiom α_9 : It is implied by the definition of the focals ABS_i and $\overline{ABS_i}$ of m_i^{a} as non empty sets in the frame of discernment Θ.

Axiom α_{10} : $\sum_{P \subseteq \Theta} m_i^{a}(P) = m_i^{a}(ABS_i) + m_i^{a}(\overline{ABS_i}) + \sum_{P \subseteq \Theta \,\&\, P \neq ABS_i \,\&\, P \neq \overline{ABS_i}} m_i^{a}(P) = e_j + (1 - e_j) + 0 = 1$ □

Theorem 9: *The function* $m_{ij}^{ncs}(P)$ *is a basic probability assignment.*

Proof: *Axiom* α_8 : By its definition, $m_{ij}^{ncs}(P) \ge 0$. Also, since $(S[\#i] - S[\#j]) \cap (S[\#j] - S[\#i]) = \emptyset$ and $|S[\#i] - S[\#j]| + |S[\#j] - S[\#i]| = |(S[\#i] - S[\#j]) \cup (S[\#j] - S[\#i])| \le |S[\#i] \cup S[\#j]|$ or equivalently, $m_{ij}^{ncs}(P) \le 1$.

Axiom α_9: It is implied by the definition of the focals $\overline{M}_j$ and Θ of m_{ij}^{ncs} as non empty sets in the frame of discernment Θ.

Axiom α_{10} : $\sum_{P \subseteq \Theta} m_{ij}^{ncs}(P) = m_{ij}^{ncs}(\overline{M}_j) + m_{ij}^{ncs}(\Theta) + \sum_{P \subseteq \Theta \,\&\, P \neq \overline{M}_j \,\&\, P \neq \Theta} m_{ij}^{ncs}(P) = d_{ij} + (1 - d_{ij}) + 0 = 1$ □

Theorem 10: *The function* $m_{ij}^{ncr}(P)$ *is a basic probability assignment.*

Proof: *Axiom* α_8: By its definition $m_{ij}^{ncr}(P) \geq 0$. Also, $|(S[\#i]-R[\#i]) \cap R[\#j]| + |(S[\#j]-R[\#j]) \cap R[\#i]| \leq |R[\#j]-R[\#i]| + |R[\#i]-R[\#j]| \leq |R[\#i] \cup R[\#j]|$ or equivalently, $m_{ij}^{ncr} \leq 1$.

Axiom α_9 : It is implied by the definition of the focals $\overline{M}_j$ and Θ of m_{ij}^{ncr} as non empty sets in the frame of discernment Θ.

Axiom α_{10} : $\sum_{P \subseteq \Theta} m_{ij}^{ncr}(P) = m_{ij}^{ncr}(\overline{M}_j) + m_{ij}^{ncr}(\Theta) + \sum_{P \subseteq \Theta \,\&\, P \neq \overline{M}_j \,\&\, P \neq \Theta} m_{ij}^{ncr}(P) = r_{ij} + (1 - r_{ij}) + 0 = 1$ □

Lemma 6: *The basic probability assignments* $m_i^{ch}, m_i^{a}, m_{ij}^{ncs}$ and m_{ij}^{ncr} *can be combined into total beliefs in any possible order*.

Proof: The focals of $m_i^{ch}, m_i^{a}, m_{ij}^{ncs}$ and m_{ij}^{ncr} are: $Cr(m_i^{ch}) \equiv CH_i \cup \overline{CH_i} = \Theta$ $Cr(m_i^{a}) \equiv AB_i \cup \overline{AB_i} = \Theta$ $Cr(m_{ij}^{ncs}) = Cr(m_{ij}^{ncr}) i \equiv \overline{M}_{ij} \cup \Theta = \Theta, \forall \; j = 1,...,n$. Therefore, their intersection is not empty and by theorem 3.4 in ([57] page 63) $m_i^{ch}, m_i^{a}, m_{ij}^{ncs}$ and m_{ij}^{ncr} can be combined in any possible order. □

Lemma 7: *It holds that* $\sum_{I \subseteq \{1,2,...,n\}} \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) = 1$ if $0 \leq x_j \leq 1$ and $\sum_{j=1}^{n} x_j = 1$

Proof(by induction on n):

Base Case(n=2): $\sum_{I \subseteq \{1,2\}} \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) = (1 - x_1)(1 - x_2) + x_1(1 - x_2) + (1 - x_1)x_2 + x_1 x_2 = 1$

Induction hypothesis(n = k): $\sum_{I \subseteq \{1,2,...,k\}} \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) = 1$

Induction step(n = k + 1):

$$\sum_{I \subseteq \{1,2,...,k,k+1\}} \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) =$$

$$\sum_{I \subseteq \{1,2,...,k\}} \prod_{j \,\varepsilon\, I \cup \{k+1\}} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) + \sum_{I \subseteq \{1,2,...,k\}} \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c \cup \{k+1\}} (1 - x_j)) =$$

$$\sum_{I \subseteq \{1,2,...,k\}} x_{k+1} \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) + \sum_{I \subseteq \{1,2,...,k\}} (1 - x_{k+1}) \prod_{j \,\varepsilon\, I} x_j \prod_{j \,\varepsilon\, I^c} (1 - x_j) =$$

$$x_{k+1}\sum_{I\subseteq\{1,2,\dots,k\}}\prod_{j\,\varepsilon\,I}x_j\prod_{j\,\varepsilon\,I^c}(1-x_j)+(1-x_{k+1})\sum_{I\subseteq\{1,2,\dots,k\}}\prod_{j\,\varepsilon\,I}x_j\prod_{j\,\varepsilon\,I^c}(1-x_j)=x_{k+1}+1-x_{k+1}=1\ \square$$

Lemma 8: *The combination of the basic probability assignments m_{ij}^{ncs} results into the following combined basic probability assignment:*

$$m^{ncs}\Big(\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,n\}\ \&\ I\neq\emptyset}\overline{M}_j\Big)=\prod_{j\,\varepsilon\,I}d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij})\ \text{and}\ m^{ncs}(\Theta)=\prod_{j=1}^{n}(1-d_{ij})$$

Proof(by induction on n): *Base Case(n=2):* None of the pairwise intersections of the focals of functions $m_{i1}^{ncs}, m_{i2}^{ncs}$ is empty(i.e. $\overline{M}_1\cap\overline{M}_2\neq\emptyset$, $\overline{M}_1\cap\Theta=\overline{M}_1\neq\emptyset$, $\Theta\cap\overline{M}_2=\overline{M}_2\neq\emptyset$, $\Theta\cap\Theta=\Theta\neq\emptyset$). Thus, their combination, according to the rule of the orthogonal sum(cf. appendix 1) (where $k_0=0$) gives:

$m^{ncs}(\overline{M}_1\cap\overline{M}_2)=m_{i1}^{ncs}(\overline{M}_1)m_{i2}^{ncs}(\overline{M}_2)=d_{i1}d_{i2}$ $\quad m^{ncs}(\overline{M}_1\cap\Theta)=m_{i1}^{ncs}(\overline{M}_1)m_{i2}^{ncs}(\Theta)=d_{i1}(1-d_{i2})$

$m^{ncs}(\Theta\cap\overline{M}_2)=m_{i1}^{ncs}(\Theta)m_{i2}^{ncs}(\overline{M}_2)=(1-d_{i1})d_{i2}$ and, $m^{ncs}(\Theta\cap\Theta)=m_{i1}^{ncs}(\Theta)m_{i2}^{ncs}(\Theta)=(1-d_{i1})(1-d_{i2})$

Induction hypothesis(n = k):

$$m'^{ncs}\Big(\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k\}\ \&\ I\neq\emptyset}\overline{M}_j\Big)=\prod_{j\,\varepsilon\,I}d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij})\ \text{and,}\ m'^{ncs}(\Theta)=\prod_{j=1}^{k}(1-d_{ij})$$

Induction step(n = k + 1): The pairwise intersections are:

$$\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k\}\ \&\ I\neq\emptyset}\overline{M}_j\cap\overline{M}_{k+1}=\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k+1\}\ \&\ I\neq\emptyset}\overline{M}_j\neq\emptyset\quad\Theta\cap\overline{M}_{k+1}=\overline{M}_{k+1}\neq\emptyset$$

$$\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k\}\ \&\ I\neq\emptyset}\overline{M}_j\cap\Theta=\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k\}\ \&\ I\neq\emptyset}\overline{M}_j\neq\emptyset\quad\Theta\cap\Theta=\Theta\neq\emptyset$$

Thus, by applying the rule of the orthogonal sum(cf. appendix 1) we get:

$$m^{ncs}\Big(\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k+1\}\ \&\ I\neq\emptyset}\overline{M}_j\Big)=m'^{ncs}\Big(\bigcap_{j\,\varepsilon\,I',\ I'\subseteq\{1,2,\dots,k\}\ \&\ I'\neq\emptyset}\overline{M}_j\Big)m_{i\,k+1}^{ncs}(\overline{M}_{k+1})=$$

$$\prod_{j\,\varepsilon\,I'}d_{ij}\prod_{j\,\varepsilon\,I'^c}(1-d_{ij})d_{i\ k+1}=\prod_{j\,\varepsilon\,I'\cup\{k+1\}}d_{ij}\prod_{j\,\varepsilon\,I'^c}(1-d_{ij})=\prod_{j\,\varepsilon\,I}d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij})$$

$$m^{ncs}\Big(\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\dots,k\}\ \&\ I\neq\emptyset}\overline{M}_j\Big)=m'^{ncs}\Big(\bigcap_{j\,\varepsilon\,I',\ I'\subseteq\{1,2,\dots,k\}\ \&\ I'\neq\emptyset}\overline{M}_j\Big)m_{i\,k+1}^{ncs}(\Theta)=$$

$$\prod_{j\,\varepsilon\,I'}d_{ij}\prod_{j\,\varepsilon\,I'^c}(1-d_{ij})(1-d_{i\ k+1})=\prod_{j\,\varepsilon\,I'}d_{ij}\prod_{j\,\varepsilon\,I'^c\cup\{k+1\}}(1-d_{i\ k+1})=\prod_{j\,\varepsilon\,I}d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij})$$

$$m^{ncs}(\Theta\cap\overline{M}_{k+1})=m'^{ncs}(\Theta)m_{i\,k+1}^{ncs}(\overline{M}_{k+1})=\prod_{j=1}^{k}(1-d_{ij})d_{i\ k+1}=\prod_{j\,\varepsilon\{k+1\}}d_{ij}\prod_{j\,\varepsilon\{1,2,\dots,k\}}(1-d_{ij})$$

$m^{ncs}(\Theta\bigcap\Theta)=m'^{ncs}(\Theta)m^{ncs}_{i\,k+1}(\Theta)=\prod_{j=1}^{k}(1-d_{ij})(1-d_{i\,k+1})=\prod_{j=1}^{k+1}(1-d_{ij})$ □

Lemma 9: *The combination of the two basic probability assignments m_{ij}^{ncs} and m_{ij}^{ncr} results into the simple support function:* $m_{ij}^{ncsr}(P)=\begin{cases}1-(1-d_{ij})(1-r_{ij}) & \text{if } P=\overline{M}_j\\(1-d_{ij})(1-r_{ij}) & \text{if } P=\Theta\end{cases}$

Proof: The pairwise intersections of the focals $\overline{M}_j$ and Θ of m_{ij}^{ncs} with the focals $\overline{M}_j$ and Θ of m_{ij}^{ncr}, $\overline{M}_j\bigcap\overline{M}_j=\overline{M}_j$, $\overline{M}_j\bigcap\Theta=\overline{M}_j$, $\Theta\bigcap\overline{M}_j=\overline{M}_j$ and $\Theta\bigcap\Theta=\Theta$ are not empty. Thus, m_{ij}^{ncs} and m_{ij}^{ncr} can be combined according to the rule of the orthogonal sum ($k_0=0$, cf. appendix 1) as: $m_{ij}^{ncsr}(\overline{M}_j)=m_{ij}^{ncr}(\overline{M}_j)m_{ij}^{ncs}(\overline{M}_j)+m_{ij}^{ncr}(\overline{M}_j)m_{ij}^{ncs}(\Theta)+m_{ij}^{ncr}(\Theta)m_{ij}^{ncs}(\overline{M}_j)=1-((1-r_{ij})(1-d_{ij}))$ and $m_{ij}^{ncsr}(\Theta)=m_{ij}^{ncr}(\Theta)m_{ij}^{ncs}(\Theta)=(1-r_{ij})(1-d_{ij})$ □

Lemma 10: *The combination of the basic probability assignments m_{ij}^{ncsr} obtained in lemma 8 results into the following basic probability assignment:*

$$m^{ncsr}\Big(\bigcap_{j\,\varepsilon\,I,\;I\subseteq\{1,2,\dots,n\}}\overline{M}_j\Big)=\prod_{j\,\varepsilon\,I}(1-(1-d_{ij})(1-r_{ij}))\prod_{j\,\varepsilon\,I^c}(1-d_{ij})(1-r_{ij}) \text{ and,}$$

$$m^{ncsr}(\Theta)=\prod_{j=1}^{n}(1-d_{ij})(1-r_{ij})$$

Proof: It can be obtained exactly as the proof of *lemma 8* by substituting m^{ncs} with m^{ncsr}, m_{ij}^{ncs} with m_{ij}^{ncsr} and d_{ij} with $(1-(1-d_{ij})(1-r_{ij}))$ □

Lemma 11: *a) The combination of the basic probability assignments m_{ij}^{ncs} , $j=1,2,\cdots,n$ provides total belief to the non determinance of an attribute i, $\overline{DET_i}$ estimated by:* $Bel_1(\overline{DET_i})=\prod_{j=1}^{n}d_{ij}$

b) The combination of the basic probability assignments m_{ij}^{ncs} and m_{ij}^{ncr}, $j=1,2,\cdots,n$ provides total belief to the non determinance of an attribute j, $\overline{DET_i}$ estimated by: $Bel_2(\overline{DET_i})=\prod_{j=1}^{n}(1-(1-r_{ij})(1-d_{ij}))$

Proof: a) $Bel_1(\overline{DET_i})=Bel_1(\bigcap_{j=1}^{n}\overline{M}_j)=\sum_{X\subseteq\bigcap_{j=1}^{n}\overline{M}_j}m^{ncs}(X)=m^{ncs}(\bigcap_{j=1}^{n}\overline{M}_j)$ and by *lemma 8* $Bel_1(\overline{DET_i})=\prod_{j=1}^{n}d_{ij}$.

b) $Bel_2(\overline{DET_i})=Bel_2(\bigcap_{j=1}^{n}\overline{M}_j)=\sum_{X\subseteq\bigcap_{j=1}^{n}\overline{M}_j}m^{ncsr}(X)=m^{ncsr}(\bigcap_{j=1}^{n}\overline{M}_j)$ and by *lemma 9,*

$Bel_2(\overline{DET_i})=\prod_{j=1}^{n}(1-(1-r_{ij})(1-d_{ij}))$ □

Theorem 11: *The combination of the basic probability assignments m_i^{a},m_i^{ch}andm^{ncs} results into the following Dempster-Shafer combined beliefs*:

$$Bel(DOM_i)=c_i e_u \quad Bel(\overline{DOM_i})=(1-c_i e_u)\prod_{j=1}^{n} d_{ij} \quad P^*(DOM_i)=1-(1-c_i e_u)\prod_{j=1}^{n} d_{ij}$$

where $u = ODC_{\#i}$.

Proof: By *lemma 6*, m_i^a, m_i^{ch} and m^{ncs} can be combined in any possible order.

Step 1(combination of m_i^a, m_i^{ch}): The pairwise intersections of the focals of m_i^a and m_i^{ch} are not empty. Consequently, their combination by the rule of the orthogonal sum ($k_0 = 0$, cf. appendix 1) gives:

$$m'(ABS_i \cap CH_i)=m_i^a(ABS_i)m_i^{ch}(CH_i)=e_u c_i \quad m'(\overline{ABS_i} \cap CH_i)=m_i^a(\overline{ABS_i})m_i^{ch}(CH_i)=(1-e_u)c_i$$

$$m'(ABS_i \cap \overline{CH_i})=m_i^a(ABS_i)m_i^{ch}(\overline{CH_i})=e_u(1-c_i) \quad m'(\overline{ABS_i} \cap \overline{CH_i})=m_i^a(\overline{ABS_i})m_i^{ch}(\overline{CH_i})=(1-e_u)(1-c_i)$$

Step 2(combination of m' , m^{ncs}): The focals of m^{ncs} and m' (obtained in the previous step) have non empty intersections. Thus, by applying of the rule of the orthogonal sum ($k_0 = 0$, cf. appendix 1):

$$m''(ABS_i \cap CH_i \bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)=m'(ABS_i \cap CH_i)m^{ncs}(\bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)= c_i e_u \prod_{j \,\varepsilon\, I} d_{ij} \prod_{j \,\varepsilon\, I^c} (1-d_{ij})$$

$$m''(\overline{ABS_i} \cap CH_i \bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)= m'(\overline{ABS_i} \cap CH_i)m^{ncs}(\bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)=$$

$$c_i(1-e_u)\prod_{j \,\varepsilon\, I} d_{ij} \prod_{j \,\varepsilon\, I^c} (1-d_{ij})$$

$$m''(ABS_i \cap \overline{CH_i} \bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)= m'(ABS_i \cap \overline{CH_i})m^{ncs}(\bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)=$$

$$(1-c_i)e_u \prod_{j \,\varepsilon\, I} d_{ij} \prod_{j \,\varepsilon\, I^c} (1-d_{ij})$$

$$m''(\overline{ABS_i} \cap \overline{CH_i} \bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)= m'(\overline{ABS_i} \cap \overline{CH_i})m^{ncs}(\bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\ldots,n\} I \neq \emptyset} \overline{M}_j)=$$

$$(1-c_i)(1-e_u)\prod_{j \,\varepsilon\, I} d_{ij} \prod_{j \,\varepsilon\, I^c} (1-d_{ij})$$

$$m''(ABS_i \cap CH_i \cap \Theta)=m'(ABS_i \cap CH_i)m^{ncs}(\Theta)=c_i e_u \prod_{j=1}^{n}(1-d_{ij})$$

$$m''(\overline{ABS_i} \cap CH_i \cap \Theta)=m'(\overline{ABS_i} \cap CH_i)m^{ncs}(\Theta)=c_i(1-e_u)\prod_{j=1}^{n}(1-d_{ij})$$

$$m''(ABS_i \cap \overline{CH_i} \cap \Theta)=m'(ABS_i \cap \overline{CH_i})m^{ncs}(\Theta)=(1-c_i)e_u \prod_{j=1}^{n}(1-d_{ij})$$

$$m''(\overline{ABS_i} \cap \overline{CH_i} \cap \Theta)=m'(\overline{ABS_i} \cap \overline{CH_i})m^{ncs}(\Theta)=(1-c_i)(1-e_u)\prod_{j=1}^{n}(1-d_{ij})$$

By accumulating these basic probability assignments into beliefs according to formula (f1) (cf. appendix 1) we get:

i. $Bel(DOM_i)=\sum_{X\subseteq(ABS_i\cap CH_i)\cup DET_i} m''(X)=$

$\sum_{I\subseteq\{1,2,\ldots,n\}\ \&\ I\neq\emptyset} m''(ABS_i\cap CH_i\bigcap_{j\,\varepsilon\,I}\overline{M}_j)+m''(ABS_i\cap CH_i\cap\Theta)=$

$c_i e_u(\sum_{I\subseteq\{1,2,\ldots,n\}\ \&\ I\neq\emptyset}(\prod_{j\,\varepsilon\,I} d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij}))+\prod_{j=1}^{n}(1-d_{ij}))=c_i e_u(\sum_{I\subseteq\{1,2,\ldots,n\}}\prod_{j\,\varepsilon\,I} d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij}))$

Thus by *lemma 6* $Bel(DOM_i)=c_i e_u$.

ii. $Bel(\overline{DOM}_i)=\sum_{X\subseteq(\overline{ABS}_i\cup\overline{CH}_i)\cap\overline{DET}_i} m''(X)=$

$m''((\overline{ABS}_i\cup CH_i)\bigcap_{j=1}^{n}\overline{M}_j)+m''((ABS_i\cup\overline{CH}_i)\bigcap_{j=1}^{n}\overline{M}_j)+m''((\overline{ABS}_i\cup\overline{CH}_i)\bigcap_{j=1}^{n}\overline{M}_j)=$

$(1-e_u)c_i\prod_{j=1}^{n} d_{ij}+e_u(1-c_i)\prod_{j=1}^{n} d_{ij}+(1-e_u)(1-c_i)\prod_{j=1}^{n} d_{ij}=(1-c_i e_u)\prod_{j=1}^{n} d_{ij}$.

Therefore, $P^*(DOM_i)=1-(1-c_i e_u)\prod_{j=1}^{n} d_{ij}$ □

Theorem 12: *The combination of the basic probability assignments* m_i^a, m_i^{ch} *with the basic probability assignments* m_{ij}^{ncs} and m_{ij}^{ncr} *results into the following Dempster- Shafer combined beliefs:*

$Bel(DOM_i)=c_i e_u \quad Bel(\overline{DOM}_i)=(1-c_i e_u)\prod_{j=1}^{n}(1-(1-r_{ij})(1-d_{ij}))$

$P^*(DOM_i)=1-(1-c_i e_u)\prod_{j=1}^{n}(1-(1-r_{ij})(1-d_{ij}))$ *where* $u=ODC_{\#i}$.

Proof: As shown in *theorem 11*, the basic probability assignments m_i^{ch} and m_i^a can be combined providing belief to the following focals: $m'(ABS_i\cap CH_i)=e_u c_i$ $m'(\overline{ABS}_i\cap CH_i)=(1-e_u)c_i$ $m'(ABS_i\cap\overline{CH}_i)=e_u(1-c_i)$ and $m'(\overline{ABS}_i\cap\overline{CH}_i)=(1-e_u)(1-c_i)$

Notice that the intersections of these focals with the focals of m^{ncsr} are not empty. Therefore, these functions can be combined according to the rule of the orthogonal sum ($k_0=0$, cf. appendix 1) as follows:

$m''(ABS_i\cap CH_i\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\ldots,n\}\ \&\ I\neq\emptyset}\overline{M}_j)=m'(ABS_i\cap CH_i)m^{ncsr}(\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\ldots,n\}\ \&\ I\neq\emptyset}\overline{M}_j)=$

$c_i e_u\prod_{j\,\varepsilon\,I} d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij})$

$m''(\overline{ABS}_i\cap CH_i\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\ldots,n\}\ \&\ I\neq\emptyset}\overline{M}_j)=m'(\overline{ABS}_i\cap CH_i)m^{ncsr}(\bigcap_{j\,\varepsilon\,I,\ I\subseteq\{1,2,\ldots,n\}\ \&\ I\neq\emptyset}\overline{M}_j)=$

$c_i(1-e_u)\prod_{j\,\varepsilon\,I} d_{ij}\prod_{j\,\varepsilon\,I^c}(1-d_{ij})$

$$m''(ABS_i \cap \overline{CH}_i \bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\dots,n\}\ \&\ I \neq \emptyset} \overline{M}_j) = m'(ABS_i \cap \overline{CH}_i) m^{ncsr}(\bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\dots,n\}\ \&\ I \neq \emptyset} \overline{M}_j) =$$

$$(1-c_i) e_u \prod_{j \,\varepsilon\, I} d_{ij} \prod_{j \,\varepsilon\, I^c} (1-d_{ij})$$

$$m''(\overline{ABS}_i \cap \overline{CH}_i \bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\dots,n\}\ \&\ I \neq \emptyset} \overline{M}_j) = m'(\overline{ABS}_i \cap \overline{CH}_i) m^{ncsr}(\bigcap_{j \,\varepsilon\, I,\ I \subseteq \{1,2,\dots,n\}\ \&\ I \neq \emptyset} \overline{M}_j) =$$

$$(1-c_i)(1-e_u) \prod_{j \,\varepsilon\, I} d_{ij} \prod_{j \,\varepsilon\, I^c} (1-d_{ij})$$

$$m''(ABS_i \cap CH_i \cap \Theta) = m'(ABS_i \cap CH_i) m^{ncsr}(\Theta) = c_i e_u \prod_{j=1}^{n} (1-d_{ij})$$

$$m''(\overline{ABS}_i \cap CH_i \cap \Theta) = m'(\overline{ABS}_i \cap CH_i) m^{ncsr}(\Theta) = c_i (1-e_u) \prod_{j=1}^{n} (1-d_{ij})$$

$$m''(ABS_i \cap \overline{CH}_i \cap \Theta) = m'(ABS_i \cap \overline{CH}_i) m^{ncsr}(\Theta) = (1-c_i) e_u \prod_{j=1}^{n} (1-d_{ij})$$

$$m''(\overline{ABS}_i \cap \overline{CH}_i \cap \Theta) = m'(\overline{ABS}_i \cap \overline{CH}_i) m^{ncsr}(\Theta) = (1-c_i)(1-e_u) \prod_{j=1}^{n} (1-d_{ij})$$

By accumulating these basic probability assignments into beliefs according to formula (f1) (cf. appendix 1) we get:

i. $Bel(DOM_i) = \sum_{X \subseteq (ABS_i \cap CH_i) \cup DET_i} m''(X) =$

$$\sum_{I \subseteq \{1,2,\dots,n\}\ \&\ I \neq \emptyset} m''(ABS_i \cap CH_i \bigcap_{j \,\varepsilon\, I} \overline{M}_j) + m''(ABS_i \cap CH_i \cap \Theta) =$$

$$c_i e_u (\sum_{I \subseteq \{1,2,\dots,n\}\ \&\ I \neq \emptyset} (\prod_{j \,\varepsilon\, I} (1-(1-d_{ij})(1-r_{ij}))) \prod_{j \,\varepsilon\, I^c} (1-d_{ij})(1-r_{ij})) + \prod_{j=1}^{n} (1-d_{ij})(1-r_{ij})) =$$

$$c_i e_u (\sum_{I \subseteq \{1,2,\dots,n\}} (\prod_{j \,\varepsilon\, I} (1-(1-d_{ij})(1-r_{ij})) \prod_{j \,\varepsilon\, I^c} ((1-d_{ij})(1-r_{ij}))))$$

Thus by *lemma 7* $Bel(DOM_i) = c_i e_u$.

ii. $Bel(\overline{DOM}_i) = \sum_{X \subseteq (\overline{ABS}_i \cup \overline{CH}_i) \cap \overline{DET}_i)} m(X) =$

$$m''((\overline{ABS}_i \cup CH_i) \bigcap_{j=1}^{n} \overline{M}_j) + m''((ABS_i \cup \overline{CH}_i) \bigcap_{j=1}^{n} \overline{M}_j) + m''((\overline{ABS}_i \cup \overline{CH}_i) \bigcap_{j=1}^{n} \overline{M}_j) =$$

$$(1-e_u) c_i \prod_{j=1}^{n} (1-(1-d_{ij})(1-r_{ij})) + e_u (1-c_i) \prod_{j=1}^{n} (1-(1-d_{ij})(1-r_{ij})) +$$

$$(1-e_u)(1-c_i) \prod_{j=1}^{n} (1-(1-d_{ij})(1-r_{ij})) = (1-c_i e_u) \prod_{j=1}^{n} (1-(1-d_{ij})(1-r_{ij}))$$

Therefore, $P^*(DOM_i) = 1-(1-c_i e_u)\prod_{j=1}^{n}(1-(1-d_{ij})(1-r_{ij}))$ □

Theorem 13: *The selection of the minimum distance isomorphism in computing function D_4 is a weighted bipartite graph matching problem.*

Proof: Since semantic homogeneity is an equivalence relation(see theorem 3), it decomposes each of the sets of the attributes of two objects #i and #j, $A_{\#i}$ and $A_{\#j}$, into the following two partitions:

$$A_{\#i}^{(1)}, \cdots, A_{\#i}^{(n)} \;\; where \;\; A_{\#i}^{(p)} \cap A_{\#i}^{(q)} = \varnothing \quad \forall \;\; p,q \;\; 1 \le p,q \le n$$

$$A_{\#j}^{(1)}, \cdots, A_{\#j}^{(m)} \;\; where \;\; A_{\#j}^{(p)} \cap A_{\#j}^{(q)} = \varnothing \quad \forall \;\; p,q \;\; 1 \le p,q \le m$$

Let $E_{\#i}$ and $E_{\#j}$ be two sets with elements all the sets $A_{\#i}^{(p)}$ and $A_{\#j}^{(q)}$, respectively. Then, we can define an isomorphism R between $E_{\#i}$ and $E_{\#j}$ as:

$$R = \left\{ (X,Y) \,|\, (X \in E_{\#i}) \text{ and } (Y \in E_{\#j}) \text{ and } (\exists\; x,y : (x \in X) \text{ and } (y \in Y) \text{ and } sh(x,y) \right\}$$

R is an isomorphism since for any two pairs of elements (X,Y), (X,Z) and (W,Z), (V,Z) belonging to it, $Y=Z$ and $W=Z$. In fact, suppose that $Y \ne Z$ and in particular that there exists an element y of Y that does not also belong to Z. Notice that y is semantically homogeneous to every element x of X(i.e. $sh(y,x) \;\forall\; x \in X$). However, since (X,Z) also belongs to R, each element x of X is semantically homogeneous to all the elements of Z (i.e. $sh(x,z) \;\forall\; z \in Z$). Then, y is semantically homogeneous to z and thus it must be member of Z. This contradicts our initial assumption that y does not belong to Z and therefore, it must be $Y \subseteq Z$. Similarly it can be proved that $Z \subseteq Y$, thus $Y=Z$. The proof that $W=V$ is identical.

For each element $e=(A_{\#i}^{(e)}, A_{\#j}^{(e)})$ of R, we can define a complete bipartite graph $G^{(e)}$, with even number of nodes, as: $G^{(e)} = (A'^{(e)}_{\#i}, A'^{(e)}_{\#j}, E^{(e)})$ where

$$A'^{(e)}_{\#i} = \begin{cases} A_{\#i}^{(e)} \quad \text{if } |A_{\#i}^{(e)}| \ge |A_{\#j}^{(e)}| \\ A_{\#i}^{(e)} \cup D_{\#i}^{(e)} \;, D_{\#i}^{(e)} = \left\{ \#d_{i1}^{(e)}, \cdots, \#d_{in}^{(e)} \right\} \;\; n = |A_{\#j}^{(e)}| - |A_{\#i}^{(e)}| \quad otherwise \end{cases}$$

$$A'^{(e)}_{\#j} = \begin{cases} A_{\#j}^{(e)} \quad \text{if } |A_{\#j}^{(e)}| \ge |A_{\#i}^{(e)}| \\ A_{\#j}^{(e)} \cup D_{\#j}^{(e)} \;, D_{\#j}^{(e)} = \left\{ \#d_{j1}^{(e)}, \cdots, \#d_{jm}^{(e)} \right\} \;\; m = |A_{\#i}^{(e)}| - |A_{\#j}^{(e)}| \quad otherwise \end{cases}$$

and $E^{(e)} = \; A'^{(e)}_{\#j} \; x \; A'^{(e)}_{\#j}$.

$D_{\#i}^{(e)}$ and $D_{\#j}^{(e)}$ are sets with special identifiers of dummy attributes. Also, $|A'^{(e)}_{\#i}| = |A'^{(e)}_{\#j}|$ and $A'^{(e)}_{\#i} \cap A'^{(e)}_{\#j} = \varnothing$ since the sets $A_{\#i}^{(e)}$ and $A_{\#j}^{(e)}$ include attributes from different objects.

The set of edges $E^{(e)}$ is partitioned into the sets

$$E_{ho}^{(e)} = A_{\#i}^{(e)} \ x \ A_{\#j}^{(e)} \ , \ E_{di}^{(e)} = A_{\#i}^{(e)} \ x \ D_{\#j}^{(e)} \ \text{and} \ E_{dj}^{(e)} = D_{\#i}^{(e)} \ x \ A_{\#j}^{(e)}$$

including the edges expressing all the possible mappings between the semantically homogeneous attributes in $A_{\#i}^{(e)}$ and $A_{\#j}^{(e)}$, connecting the attributes of #i with dummy attributes of #j(expressing mappings of the former attributes to *null* attributes) and connecting the attributes of #j with dummy attributes of #i (expressing mappings of the former attributes to *null* attributes), respectively.

Furthermore, each edge (#v,#u) in $E^{(e)}$ is assigned a weight $c_{\#v\#u}$ defined as

$$c_{\#v\#u} = \begin{cases} d'(\#v,\#u)s(\#v)s(\#u) & \text{if } (\#v,\#u) \ \varepsilon \ E_{ho}^{(e)} \\ s(\#v)^2 & \text{if } (\#v,\#u) \ \varepsilon \ E_{di} \\ s(\#u)^2 & \text{if } (\#v,\#u) \ \varepsilon \ E_{dj} \end{cases}$$

The factors $d'(\#v,\#u)$, $s(\#v)$, $s(\#u)$ are defined by definition 14. Then, the weighted bipartite graph matching problem is to find a matching $M^{(e)}$ (i.e. a susbset of $E^{(e)}$) with the minimum possible sum of weights.

In an algebraic formulation, let the variables $x_{\#v\#u}$ ($\#v \ \varepsilon A'^{(e)}_{\#i}$ and $\#u \ \varepsilon \ A'^{(e)}_{\#j}$) denote whether or not an edge belongs to a matching of $G^{(e)}$ by taking the values 1 and 0, respectively. The weighted bipartite graph matching problem is to find an assignment to the variables $x_{\#v\#u}$, minimizing the objective function

$$\sum_{\#v \ \varepsilon \ A'^{(e)}_{\#i} \ , \ \#u \ \varepsilon \ A'^{(e)}_{\#j}} x_{\#v\#u} \, c_{\#v\#u} \tag{F}$$

while satisfying the constraints

$$\sum_{\#u \ \varepsilon \ A'^{(e)}_{\#j}} x_{\#v\#u} = 1 \ \ \forall \ \ \#v \ \varepsilon \ A'^{(e)}_{\#i} \tag{c1}$$

$$\sum_{\#v \ \varepsilon \ A'^{(e)}_{\#i}} x_{\#v\#u} = 1 \ \ \forall \ \ \#u \ \varepsilon \ A'^{(e)}_{\#j} \tag{c2}$$

and

$$x_{\#v\#u} = 0 \ or \ 1 \forall \ \ \#v,\#u : \#v \ \varepsilon \ A'^{(e)}_{\#i} \ \text{and} \ \#u \ \varepsilon \ A'^{(e)}_{\#j} \tag{c3}$$

The constraint c1(c2) together with c3 guarantee that elements of $A'^{(e)}_{\#i}$ ($A'^{(e)}_{\#j}$) can be mapped onto only one element of $A'^{(e)}_{\#j}$ ($A'^{(e)}_{\#i}$). Also the minimization of function (F) guarantees that the isomorphism between the elements of $A'^{(e)}_{\#i}$ and $A'^{(e)}_{\#j}$ will have the lowest possible overall attribute distance, due to the particular assignment of factors $c_{\#v\#u}$. It is known that even if (c3) is relaxed to $x_{\#v\#u} \geq 0$, the resulting linear programming problem is of a special form that guarantees the existence of integer(therefore, 0 or 1) optimal solutions.

Notice that any attribute in $A_{\#i}^{(e)}$ can only be mapped onto one attribute of $A_{\#j}^{(e)}$(and vice versa) or to the null attribute. Also the weights of the edges in $E^{(e)}$ are exactly the distance factors between semantically homogeneous attributes(for edges in $E_{ho}^{(e)}$) or unmapped attributes(for edges in $E_{di}^{(e)} \cup E_{dj}^{(e)}$) defined by definition 19. Thus, the optimal matching of the bipartite graph $G^{(e)}$ will also be the optimal isomorphism between the attributes $A_{\#i}^{(e)}$ and $A_{\#j}^{(e)}$.

Therefore, solving the bipartite weighted graph matching problem for all the graphs $G^{(e)}$ defined by R and taking the union of the obtained optimal matchings, we obtain the optimal isomorphism between the attributes of the objects #i and #j. Attributes belonging to those sets of $E_{\#i}$ and $E_{\#j}$ which are not included in R cannot be mapped to any attribute of the other object and their contribution to the attribution distance is fixed(i.e. equal to the square of their saliences). These factors must be added to the sum of the minimal distances obtained by solving the bipartite weighted graph matching problems for the graphs $G^{(e)}$ □

References

1. Abiteboul S., Hall R., *IFO: A Formal Semantic Database Model*, ACM Transactions on Database Systems, 12(4), December 1987

2. Aha D., *Case-Based Learning Algorithms*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1991

3. Allen B., *Case-Based Reasoning: Business Applications*, Communications of the ACM, 33(3), 1994

4. Ashley K., *Modelling Legal Argument: Reasoning with Cases and Hypotheticals*, A Bradford Book, The MIT Press, 1990

5. Bourrelly L., Chouraqui E., *A Formal Approach to Analogical Reasoning*, Approximate Reasoning in Expert Systems, Elsevier Science Publishers, 1985

6. Borgida A., et al., *Generalization/Specialization as a Basis for Software Specification*, On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages, (ed) Mylopoulos J., Springer-Verlang, 1986

7. Burstein M., *Concept Formation By Incremental Analogical Reasoning and Debugging*, Machine Learning: An Artificial Intelligence Approach, Vol. II, Morgan Kaufmann Publishers Inc., Los Altos, California, 1986

8. Carbonell J.G., *Learning by Analogy: Formulating and Generalizing Plans from Past Experience*, Machine Learning: An Artificial Intelligence Approach, Vol. I, Morgan Kaufmann Publishers Inc., Los Altos, California, 1983

9. Carbonell J.G., *Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition*, Machine Learning: An Artificial Intelligence Approach, Vol. II, Morgan Kaufmann Publishers Inc., Los Altos, California, 1986

10. Maria Christoforaki, *Cultural documentation with system CLIO*, FORTH-ICS / TR-088, Institute of Computer Science, Foundation for Research and Technology-Hellas, Iraklion, Crete, Greece, 1994

11. Constantopoulos P., Doerr M., *The Semantic Index System: A Brief Presentation*, Institute of Computer Science, Foundation for Research and Technology-Hellas, Iraklion, Crete, Greece, 1993

12. Callan J., Fawrett T., Rissland E., *CABOT: An Adaptive Approach to Case Based Search*, Proceedings of the International Joint Conference on Artificial Intelligence, 1991

13. Cook D., *The Base Selection Task in Analogical Planning*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1991

14. Cain T., et al., *Using Domain Knowledge to Influence Similarity Judgements*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1991

15. Doerr M., Klimathianakis P., *A TELOS Model for C++ Static Analysis Data*, Technical Report, Institute of Computer Science, Foundation for Research and Technology-Hellas, Iraklion, Crete, Greece, September 1993

16. Fertig S., Gelernter D., *FGP: A Virtual Machine for Acquiring Knowledge from Cases*, Proceedings of the International Joint Conference on Artificial Intelligence, 1991

17. Falkenhainer B., Forbus K., Gentner D., *The Structure Mapping Engine: Algorithm and Examples*, Artificial Intelligence, 41, 1990

18. Fikes R., Kehler T., *The Role of Frame-Based Representation in Reasoning*, Communications of the ACM, 28(9),1985

19. Furnas G.W., et al., *The Vocabulary Problem in Human-System Communication*, Communications of the ACM, 30(11), 1987

20. Gentner D., *Analogical Inference and Analogical Access*, Analogica, (ed) Armand Prieditis, Morgan Kaufmann Pub., 1988

21. Giles J.R., *Introduction to the Analysis of Metric Spaces*, Australian Mathematical Society, Cambridge University Press, 1987

22. Gentner D., Landers R., *Analogical Reminding: A Good Match is Hard to Find*, Proceedings of International Conference on Systems, Man and Cybernetics, Tucson, AZ, 1985

23. Greiner R., *Learning and Understanding by Analogies*, Artificial Intelligence, 35, 1988

24. Gentner D., Toupin C., *Systematicity and Surface Similarity in the Development of Analogy*, Cognitive Science, 1986

25. Glasser G. J., Winter R.F., *Critical Values of the Coefficient of Rank Correlation for Testing the Hypothesis of Independence*, Biometrika, 48, 1961

26. Hall R., *Computational Approaches to Analogical Reasoning: A Comparative Analysis*, Artificial Intelligence, 39, 1989

27. Hull R., King R., *Semantic Database Modelling: Survey, Applications and Research Issues*, ACM Computing Surveys, 19(3), 1987

28. Holyoak K., Thagard P., *Analogical Mapping by Constraint Satisfaction*, Cognitive Science, 13, 1989

29. Indurkhya B., *Constrained Semantic Transference: A Formal Theory of Metaphors*, Synthese, 68(3), Reidel Publishing Company, 1986

30. Kedar-Cabelli S., *Purpose-Directed Analogy*, Proceedings of the 7th Annual Conference of the Cognitive Science Society, Irvine, CA, 1985

31. Kedar-Cabelli S., *Analogy-From a Unified Perspective*, Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science and Philosophy, (ed) Helman D.H., Kluwer Academic Pub., Dordrecht, The Netherlands, 1988

32. Keen M., *Presenting Results of Experimental Retrieval Comparisons*, Information Processing & Management, 28(4), 1992

33. Kevork K., *Analysis of Regression and Correlation*, Statistical Methods, Vol. III, Economical University of Athens, Athens, 1984

34. Kowalski J., *Topological Spaces*, Academic Press, N.Y, 1965

35. Koubarakis M., et al., *Telos: Features and Formalization*, Technical Report, Technical Report Series, FORTH/CSI/TR/1989/018, Institute of Computer Science, Foundation for Research and Technology-Hellas, Iraklion, Crete, Greece, 1989

36. Kolonder J., *Judging Which is the "Best" Case for a Case-Based Reasoner*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1988

37. Kolonder J.L., et al., *A Process Model of Case-Based Reasoning in Problem Solving*, Proceedings of International Joint Conference on Artificial Intelligence(IJCAI-85), Los Angeles, CA, 19855

38. Kuiper T., *Inductive Analogy By Similarity and Proximity*, Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science and Philosophy, (ed) Helman D.H., Kluwer Academic Pub., Dordrecht, The Netherlands, 1988

39. Maiden N., *Analogical Specification Reuse During Requirements Analysis*, Ph.D Thesis, Dept. of Business Computing, City University, London , July 1992

40. Mylopoulos J., et. al., *Telos: Representing Knowledge About Information Systems*, ACM Transactions on Information Systems, 8(4), 1990

41. McDermott J., *Learning to Use Analogies*, Proceedings of the International Joint Conference on Artificial Intelligence(IJCAI-79), Tokyo, 1979

42. Meyer B., *Object Oriented Software Construction*, C.A.R Hoare, Series Editor, 1989

43. Mostow J, *Design By Derivational Analogy: Issues in the Automatic Replay of Design Plans*, Artificial Intelligence, 40, 1989

44. Maiden N., Sutcliffe A., *Exploiting Reusable Specifications through Analogy*, Communications of the ACM, 35(4), 1992

45. Munyer C., *Analogy as a Means of Discovery in Problem Solving and Learning*, Ph.D Thesis, University of California at Santa Cruz, CA, 1981

46. Niiniluoto I., *Analogy and Similarity in Scientific Reasoning*, Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science and Philosophy, (ed) Helman D.H., Kluwer Academic Pub., Dordrecht, The Netherlands, 1988

47. Novick L., *Analogical Transfer: Processes and Individual Differences*, Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science and Philosophy, (ed) Helman D.H., Kluwer Academic Pub., Dordrecht, The Netherlands, 1988

48. Porter W.B., et al., *Concept Learning and Heuristic Classification in Weak-Theory Domains*, Artificial Intelligence, 45, 1990

49. Peckham J., Maryanski F., *Semantic Data Models*, ACM Computing Surveys, 20(3), 1988

50. Papadimitriou C., Steiglitz K., *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982

51. Russel S., *Analogy By Similarity*, Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science and Philosophy, (ed) Helman D.H., Kluwer Academic Pub., Dordrecht, The Netherlands, 1988

52. Spanoudakis G., Constantopoulos P., *Similarity for Analogical Software Reuse: A Conceptual Modelling Approach*, Proceedings of the 5th International Conference on Advanced Information Systems Engineering, (CAiSE ’93), (eds) Rolland C., Cauvet C., LNCS 685, June 1993

53. Spanoudakis G., Constantopoulos P., *Measuring Similarity Between Software Artifacts*, Proceedings of the 6th International Conference on Software Engineering and Knowledge Engineering, SEKE ’94, Printed by Knowledge Systems Institute, Skokie, 1994

54. Spanoudakis G., Constantopoulos P., *Similarity for Analogical Software Reuse: A Computational Model*, Proceedings of the 11th European Conference on Artificial Intelligence(ECAI ’94), Amsterdam, The Netherlands, August 1994

55. Spanoudakis G., Constantopoulos P., *On Evidential Feature Salience*, Proceedings of the 5th International Conference on Database and Expert Systems Applications(DEXA’94), Athens, September 1994

56. Seifert C., Hammond C., *Why There is No Analogical Transfer*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1988

57. Shafer G., *A Mathematical Theory of Evidence*, Princeton University Press, 1975

58. Sjoberg L., *A Cognitive Theory of Similarity*, Goteborg Psychological Reports, Number 10, Vol. II, 1972

59. Sycara K., Navinhandria D., *Influences: A Thematic Abstraction for Creative Use of Multiple cases*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1991

60. van Someren M.W., *Using Attribute Dependencies for Rule Learning*, European Workshop on Knowledge Representation and Organization in Machine Learning, Springer- Verlang, 1988

61. Spanoudakis G., *Analogical Similarity of Objects: A Conceptual Modelling Approach*, Doctoral Dissertation, Department of Computer Science, University of Crete, Heraklion, Crete, Greece,

September 1994 (also available from *http://web.cs.city.ac.uk/homes/gespan/info.html*)

62. Stroustrup B., *The C++ Programming Language*, Addison-Wesley, Reading, Mass. , 1987

63. Su L., *Evaluation Measures for Interactive Information Retrieval*, Information Processing and Management, 28(4), 1992

64. Tausend B., Bell S., *Analogical Reasoning for Logic Programming*, Proceedings of European Working Session on Learning, (ed) Kodratoff Y., Springer-Verlang, 1991

65. Thagard P. et al., *Analog Retrieval by Constraint Satisfaction*, Artificial Intelligence, 46, 1990

66. Turner M., *Categories and Analogies*, Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science and Philosophy, (ed) Helman D.H., Kluwer Academic Pub., Dordrecht, The Netherlands, 1988

67. Tversky A., *Features of Similarity*, Psychological Review, 44(4), July 1977

68. Yamane T., Kinti A., *Mathematics for Economists*, Vol. I and II, Gutenberg, Athens, 1980

69. Veloso M., Carbonell J., *Learning by Analogical Replay in PRODIGY: First Results*, Proceedings of European Working Session on Machine Learning, (ed) Kodratoff Y., Springer-Verlang, 1991

70. Weber M., *Implementation Models for the SIB*, ITHACA.SNI.92.E2.#2, Siemens-Nixdorf, September, 1992

71. Wegner P., *Dimensions of Object Based Language Design*, Proceedings of the ACM Conference on Object-Oriented Programming, Systems, Languages and Applications, (OOPSLA ’87), 1987

72. Winston P., *Learning and Reasoning By Analogy*, Communications of the ACM 23(12), 1980

73. Whitaker L., et al., *Using Qualitative or Multi-Attribute Similarity to Retrieve Useful Cases from a Case Base*, Proceedings of the DARPA Workshop on Case-Based Reasoning, 1988

74. Spanoudakis G., Similarity Analyzer: An Implementation Overview, Working Paper No 11, Institute of Computer Science, Foundation of Research and Technology-Hellas, September 1994 (also available from *http://web.cs.city.ac.uk/homes/gespan/info.html*)

75. Maiden N., et al., *Computational Mechanisms for Distributed Requirements Engineering*, Proceedings of the 7th International Conference on Software Engineering & Knowledge Engineering (SEKE ’95), Pitsburg, Maryland, USA, June 1995

76. Spanoudakis G., Constantopoulos P., *Integrating Specifications: A Similarity Reasoning Approach*, submitted for publication to the Automated Software Engineering Journal

77. Constantopoulos P., Spanoudakis G., *Integrating Specifications* (a position paper), Proceedings of the Dagstuhl Seminar on Systems Requirements Analysis - Management & Exploitation; Schloss Dagstuhl, Germany, October 4th-7th 1994 (also available from *http://web.cs.city.ac.uk/homes/gespan/info.html*)

78. Spanoudakis G., Constantopoulos P., Kyprioti M., *Similarity Analysis for Process Modelling of Requirements Engineering* (a position paper), Proceedings of the Dagstuhl Seminar on Systems Requirements Analysis - Management & Exploitation; Schloss Dagstuhl, Germany, October 4th-7th 1994 (also available from *http://web.cs.city.ac.uk/homes/gespan/info.html*)

79. Constantopoulos P. et al., *The Software Information Base: A Server for Reuse*, The VLDB Journal(to appear)

80. Rosch E. et al., *Basic Objects in Natural Categories*, Cognitive Psychology, 8, 1976