algorithms

Article

A Temporal Case-Based Reasoning Platform Relying on a Fuzzy Vector Spaces Object-Oriented Model and a Method to Design Knowledge Bases and Decision Support Systems in Multiple Domains

Joël Colloc [1,*], Relwendé Aristide Yameogo [1,2], Peter Summons [3], Lilian Loubet [1], Jean-Bernard Cavelier [1] and Paul Bridier [1]

[1] UMR IDEES 6266, University Le Havre Normandy, 76600 Le Havre, France; yraristide@gmail.com (R.A.Y.); lilian.loubet@univ-lehavre.fr (L.L.); jeanbernard.cavelier@gmail.com (J.-B.C.); pbridier@aol.com (P.B.)
[2] Département de Santé Publique, Université Joseph Ki-Zerbo, Ougadougou 03 BP 7021, Burkina Faso
[3] School of Information and Physical Sciences, The University of Newcastle, Callaghan 2308, Australia; peter.summons@newcastle.edu.au
* Correspondence: joel.colloc@univ-lehavre.fr

Citation: Colloc, J.; Yameogo, R.A.; Summons, P.; Loubet, L.; Cavelier, J.-B.; Bridier, P. A Temporal Case-Based Reasoning Platform Relying on a Fuzzy Vector Spaces Object-Oriented Model and a Method to Design Knowledge Bases and Decision Support Systems in Multiple Domains. *Algorithms* **2022**, *15*, 66. https://doi.org/10.3390/a15020066

Academic Editor: Edward Rolando Núñez-Valdez

Received: 23 December 2021
Accepted: 6 February 2022
Published: 19 February 2022

Abstract: Knowledge bases in complex domains must take into account many attributes describing numerous objects that are themselves components of complex objects. Temporal case-based reasoning (TCBR) requires comparing the structural evolution of component objects and their states (attribute values) at different levels of granularity. This paper provides some significant contributions to computer science. It extends a fuzzy vector space object-oriented model and method (FVSOOMM) to present a new platform and a method guideline capable of designing objects and attributes that represent timepoint knowledge objects. It shows how temporal case-based reasoning can use distances between temporal fuzzy vector functions to compare these knowledge objects' evolution. It describes examples of interfaces that have been implemented on this new platform. These include an expert's interface that describes a knowledge class diagram; a practitioner's interface that instantiates domain objects and their attribute constraints; and an end-user interface to input attribute values of the real cases stored in a domain case database. This paper illustrates resultant knowledge bases in different domains, with examples of pulmonary embolism diagnosis in medicine and decision making in French municipal territorial recomposition. The paper concludes with the current limitations of the proposed model, its future perspectives and possible platform enhancements.

Keywords: time case-based reasoning; time fuzzy vectorial space; fuzzy object-oriented design method; structural similarity; qualifying attributes; pulmonary embolism; territorial recomposition; clinical modeling; psychology; infectious diseases; decision support system; object composition multiple inheritance

1. Introduction

1.1. Contributions

The design of temporal knowledge bases in complex domains, such as medicine, territories, management, and psychology, must incorporate many attributes of different kinds from a large number and variety of objects, which are often themselves components of larger, complex objects. Temporal case-based reasoning (TCBR) is an extension of case-based reasoning (CBR) that requires a comparison of the evolution of the complex structure over time. This requires a temporal comparison of the complex structure's component objects and their states (the value of their attributes), where component objects must be considered at different levels of granularity, according to their composition in the complex structure. This paper uses a fuzzy vector space object-oriented model and method (FVSOOMM), proposed by Colloc [1]. In the ESM'2019 conference paper [2], FVSOOMM was presented and utilized to model the personality, cognitive dissonance, and emotions in a marketing case example. Colloc's paper presented the main advantages and drawbacks of fuzzy rules. It also described the advantages of FVSOOMM in assessing the evolution of a knowledge object's structure and attributes, compared to rule-based systems that rely on fuzzification/defuzzification operations and induction on a set of selected attributes. FVSOOMM extends the object-oriented model mainly based on the object–composition relationship [3]. FVSOOMM is not a fuzzy logic approach, but rather a fuzzy extension of object models that offer a more general model of knowledge than fuzzy logic [4]. Zadeh himself mentioned that object-oriented approaches offer a more general model of knowledge representation than fuzzy logic [5]. This paper presents the following contributions:

1. Extension of the TFVS model
   This paper extends the temporal fuzzy vector space (TFVS) model with the different steps of the method within FVSOOMM. This provides a new mechanism to model knowledge bases in many different and complex domains, and to capture the semantics of the relationships between the objects and actors described in the ontology of their knowledge domain. In common with other authors [6–9], the present authors trust that object-oriented approaches offer a more general model of knowledge representation than just from fuzzy logic alone.
2. Design and implementation of a platform
   The novelty of the tool is a platform that supports the method and provides three interfaces: an expert interface that allows the expert to describe the class diagram of the domain knowledge, and an attribute language that is used to define the attribute descriptors that are made available in a repository to the practitioners. A practitioners' interface allows stakeholders in the field to choose object types and to instantiate the necessary knowledge objects and their attribute descriptors that define and control the characteristics and range of attribute values of knowledge objects. The third interface is for end users, who simply enter the attribute values during their professional practice as they are solving real cases, where they are then stored in an experience case database. The case database is an experience repository that is particularly useful to practitioners, as it elaborates new research in their field. The method of FVSOOMM describes the whole process of a knowledge base design and how to develop the corresponding decision support system (DSS).
3. A method for temporal case-based reasoning
   This advances other works as case-based reasoning (CBR) based on an object-oriented model to develop DSS. The method defines the distances required to implement temporal case-based reasoning (TCBR) by comparing the time fuzzy vector functions that describe the evolution of the case objects and their attributes over time. The TCBR system sorts similar cases within an interval of time.

1.2. Structure

This paper is structured as follows. Section 1.1 outlines the main contributions of the paper. Section 2 overviews modeling knowledge and time with fuzzy logic and introduces the proposed fuzzy vector space, object-oriented model and method (FVSOOMM), a model and method to implement fuzzy vectorial spaces in an object-oriented model. Section 2.2 describes the use and calculation of fuzzy vector spaces and overviews FVSOOM, the model within FVSOOMM. Section 2.3 describes the steps of the methods used in FVSOOMM with temporal fuzzy vector spaces to produce knowledge bases and experience case bases for decision support systems. Section 2.4 describes the distance computing and the case-based reasoning cycle is overviewed and extended to temporal case-based reasoning (TCBR). The computing efficiency advantages of the proposed method for calculating distances in TCBR are presented. Section 2.5 presents two different case example applications that have been designed and implemented using FVSOOMM. The first is in the field of clinical medicine: pulmonary embolism diagnosis and follow-up. The second describes decision making in the territorial recomposition of French municipalities. Section 3 discusses the results and advantages of using FVSOOMM. It also outlines the present constraints and future work. Section 4 concludes with the two major contributions of the paper.

2. Materials and Methods

2.1. Knowledge and Temporal Modelling with Fuzzy Logic

Since fuzzy set theory was introduced by Lofti Zadeh, models have been proposed for a variety of useful applications. Many of these models rely on a logic approach based on *modus ponens* inference using fuzzy rules [10–13]. These fuzzy rules combine *membership functions* and *linguistics variables* to describe subsets of the characteristic values of specific attributes, or parameters, used to express knowledge chunks stored in knowledge bases. Fuzzy set theory constitutes a great enhancement and offers flexibility to take into consideration the uncertainty and the approximation of necessary variable features. In a 2008 paper, Zadeh described the importance of relationships in knowledge modeling: *"Fuzzy logic is much more than a logical system. It has many facets. The principal facets are: logical, fuzzy-set-theoretic, epistemic and relational. Most of the practical applications of fuzzy logic are associated with its relational facet. Fuzzy logic is viewed in a nonstandard perspective. In this perspective, the cornerstones of fuzzy logic and its principal distinguishing features are graduation, granulation, precision, and the concept of a generalized constraint"* [5]. In a similar way, this paper proposes a model that is not a logic model, but rather a fuzzy object-oriented extension, which offers fuzzy relationships to capture more semantics, time modeling, fuzzy composition (is-part-of) relationships, and object attribute and composition (has-a) relationships represented by the $\bigoplus$ operator.

2.2. Fuzzy Vector Spaces and the Object-Oriented Model (FVSOOM)

This section commences with a brief summary of fuzzy vector space (FVS) literature and then describes the fuzzy vector space model FVSOOM, within FVSOOMM, the model and method that was introduced in Section 1. Many papers concerning fuzzy vector spaces rely on extensions of fuzzy set theory and fuzzy logic as proposed by Lofti Zadeh [4,14–17]. Katzaras and Liu defined the notion of fuzzy vector space from Zadeh's fuzzy set theory [18]. Lubczonok proved the properties of the dimensions of these fuzzy vector spaces [19]. Fuzzy set theory allows combination of the values of several attributes with $\in$, AND, and OR, fuzzy operators. Fuzzy set theory has been used to implement analogical problem solving, for example, to assess the results of students in a classroom solving mathematical problems [20]. One interesting application of the use of fuzzy rules was to implement fuzzy cognitive maps to residential mobility in urban spaces, describing the relationship between attributes, causality relationships and linguistic variables [21]. However, most of these works describe domains that have only a few attributes and do not rely on semantic models, such as the entity–relationship model containing entity sets, entities, attributes, and relationships, or the object-oriented model, containing classes, objects, attributes, and associations. FVSOOM, the model of FVSOOMM, relies on a fuzzy function that represents an attribute as a vector, with a scalar value in its definition set, $[min, max]$ at each time instant t. This vector represents a force and can be combined with other vectors in a vector space using the addition of vectors and the scalar product and, more generally, all vector space properties. These properties simplify the combination of attributes and object composition that are considered vector addition operators in object-oriented models. Fuzzy vector space (FVS) is based on the following function that depicts two opposite vectors (Figure 1).

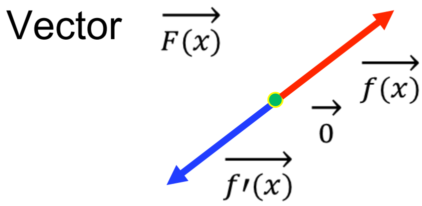

Figure 1. Vector function $F(\vec{x}) : [-1, 1] \longrightarrow [-1, 1]$.

The function is defined in Equation (4) and shown on Figure 2, where $\frac{2}{1+e^{-2x}} - 1 = tanh(x)$, which is uniformly continuous and defined on $[-1, 1]$. The limits $\forall x \in \mathbb{R}$, $\lim_{x\to-\infty} tanh(x) = -1$ *and* $\lim_{x\to+\infty} tanh(x) = 1$. Thus, according to the continuous extension theorem, $tanh(x)$ is continuous on $[-1, 1]$ in Equation (4). Equations (1)–(3), respectively, describe the positive vector $f(x)$, the negative vector $f'(x)$ and the hyperbolic tangent $F(x)$. Each attribute x_i of an object is described by a function $F(x_i$ and its corresponding vector (Figure 1). If the attribute i is changing at each time t, the fuzzy value is calculated by the function $F_{i,t}(x_{i,t})$, where i is the order number of the attribute in the object, $x_{i,t}$ is its value, and the scalar $\alpha_{i,t}$ is the weight of the attribute in the object at each instant t. The weight in the object—$\alpha_{i,t}$ or the function $F_{i,t}$—could change over the time (for example, from linear to exponential, or to logarithmic) but it is rarely needed.

$$f : [0, 1] \to [0, 1], f(x) = \frac{2}{1 + e^{-k(x-s)}} - 1 \tag{1}$$

$$f' : [-1, 0] \to [-1, 0], f'(x) = \frac{2}{1 + e^{-k(x-s)}} - 1 \tag{2}$$

$$F : [-1, 1] \to [-1, 1], F(x) = tanh(\frac{kx - s}{2}) \tag{3}$$

$$F : [-1, 1] \to [-1, 1]/F(x_i, t) = \begin{cases} x = -1 & F(x_i, t) = -1, \\ x \in [-1, 1] & F(x_i, t) = tanh(\frac{kx_i - s}{2}), \\ x = 1 & F(x_i, t) = 1. \end{cases} \tag{4}$$

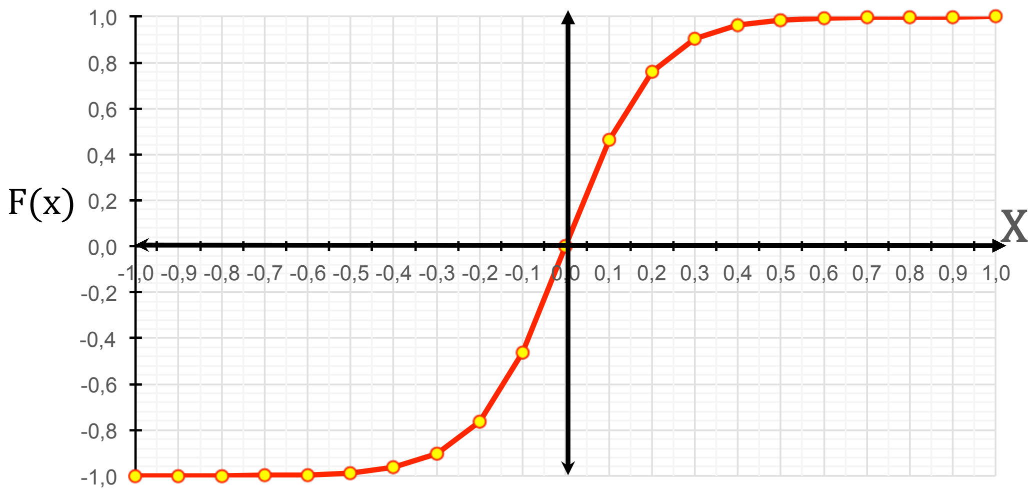

Figure 2. A sigmoid membership function on $[-1, 1] \longrightarrow [-1, 1]$.

It follows that $\forall x \in [-1,1], f(x) = -f'(-x)$. The resultant function r of fuzzy functions is described by Equation (5):

$$\forall (x_{i,t}, \ldots, x_{n,t}) \in [-1,1]^n, r(x_{i,t}, \ldots, x_{n,t}) = \sqrt{\frac{\sum_{i=1}^{n} |\alpha_{i,t}|(f_{i,t}(x_{i,t}))^2}{\sum_{i=1}^{i=n} |\alpha_{i,t}|}} \in [-1,1]. \quad (5)$$

The problem here is that Equation (5) does not correctly express cases where $f_{i,t}(x_{i,t})$ is negative and, therefore, does not allow the determination of the negative vector, as the root of a negative real is not a real but a complex number. Assume a set of x_i attributes of an object $O(x_i, \ldots, x_n)$. In practice, it is suitable to treat all x_i attributes with values higher than the center value, VC, where $f(x_i)$ is positive, to calculate the positive vector $\overrightarrow{OU^+} = \overrightarrow{O\forall x_i \geq Vc}$, and those $x_i < VC$ where $f(x_i)$ is negative in order to calculate the negative vector $\overrightarrow{OU^-} = \overrightarrow{Ox\forall x_i < VC}$, and then to combine these two vectors $\overrightarrow{OU} = \overrightarrow{OU^+} + \overrightarrow{OU^-}$. This is not a problem since the addition of vectors is commutative and associative. Note that with respect to each attribute x_i of the object O and its definition domain $DO.x_i = [Min(x_i), Max(x_i)]$, we are free to choose the value of the VC, so it can be either the average, the median, the mode, or we could use $VC = \frac{|Max(x_i)| - |Min(x_i)|}{2}$.

2.2.1. Normalization Function

We must use a normalization function $H : [Min(x_i), Max(x_i)] \rightarrow [-1,1], H(VC) = 0$, where VC corresponds to the null vector. The normalization function H is described by the following Equation (6). It is used to adapt the definition domain of attribute values to fit in the $[-1,1]$ interval:

$$H : \mathbb{R} \rightarrow [-1,1],\ H(x) = \begin{cases} -1 & if\ x = min, \\ (x-cv)/(max-cv) & if\ x \geq cv, \\ (x-cv)/(cv-min) & if\ x < cv, \\ 1 & if\ x = max \end{cases} \quad (6)$$

An example of the use of H(x) for the attribute Age is given in Table 1, for the pulmonary embolism example application. Two methods of calculating FVS are shown below. The first is expressed by Equations (7)–(10).

Table 1. Fuzzy characteristic function table for μAge(H(x)).

Age	H(x)	μAge(H(x))	Age	H(x)	μAge(H(x))
35	−1.00	−1.00	51	0.07	0.17
36	−0.93	−0.98	52	0.13	0.32
38	−0.80	−0.96	53	0.20	0.46
39	−0.73	−0.95	54	0.27	0.58
40	−0.67	−0.96	55	0.33	0.68
41	−0.60	−0.91	56	0.40	0.76
42	−0.53	−0.87	57	0.47	0.82
43	−0.47	−0.82	58	0.53	0.87
44	−0.40	−0.76	59	0.60	0.91
45	−0.33	−0.68	60	0.67	0.93
46	−0.27	−0.58	61	0.73	0.95
47	−0.20	−0.46	62	0.80	0.96
48	−0.13	−0.32	63	0.87	0.97
49	−0.07	−0.17	64	0.93	0.98
50	0.00	0.00	65	1.00	1.00

2.2.2. Method 1 of FVS Calculation

$nbTot = nbPos + nbNeg$. Assume an object O at each time instant t. $O(x_{i,t} \ldots x_{n,t})$, $\forall x_{i,t} \in [-1,1], \overrightarrow{OU_t} = \overrightarrow{OU_t^+} + \overrightarrow{OU_t^-}$.

$$O(x_{i,t} \ldots x_{n,t}), \begin{cases} x_{i,t} \geq VC & nbPos = \sum_{i=1}^{i=n} \alpha_i \\ x_{i,t} < VC & nbNeg = \sum_{i=1}^{i=n} \alpha_i \end{cases} \tag{7}$$

$$\forall x_{i,t}, \begin{cases} x_{i,t} \geq VC & sum^2Pos \leftarrow sum^2Pos + \alpha_{i,t}(f(x_{i,t}))^2, \\ x_{i,t} < VC & sum^2Neg \leftarrow sum^2Neg + \alpha_{i,t}(f(x_{i,t}))^2. \end{cases} \tag{8}$$

$$\|\overrightarrow{OU^+}\| = \sqrt{\frac{sum^2Pos}{nbPos}} \qquad \|\overrightarrow{OU^-}\| = \sqrt{\frac{sum^2Neg}{nbNeg}} \tag{9}$$

$$\|\overrightarrow{OU}\| = \begin{cases} \|\overrightarrow{OU^+}\|^2 \geq \|\overrightarrow{OU^-}\|^2, & \sqrt{\frac{\|\overrightarrow{OU^+}\|^2 + \|\overrightarrow{OU^-}\|^2}{nbTot}} \\ \|\overrightarrow{OU^-}\|^2 > \|\overrightarrow{OU^+}\|^2, & -\sqrt{\frac{\|\overrightarrow{OU^-}\|^2 - \|\overrightarrow{OU^+}\|^2}{nbTot}} \end{cases} \tag{10}$$

2.2.3. Method 2 of FVS Calculation with a Sign Function

The sign(x) function returns 1 if $x > 0$, -1 if $x < 0$ and 0 if $x = 0$. This method relies on Equations (11) and (12).

Assume an object $O(x_{i,t} \ldots x_{n,t})$,

$$\alpha_i \in \mathbb{R}^{+*}, S = \frac{\sum_{i=1}^{i=n} Sign(f(x_{i,t})).\alpha_i.(f(x_{i,t}))^2}{\sum_{i=1}^{i=n} \alpha_i} \tag{11}$$

$$\|\overrightarrow{Ou}\| = Sign(S).\sqrt{Sign(S).S} \tag{12}$$

Method 2 is easier to implement and calculate the FVS, as seen by the Java listings in Section 2.3.

2.3. FVSOOMM: A Model and Method to Implement Fuzzy Vector Spaces in an Object-Oriented Model

Several artificial intelligence (AI) methods were proposed in order to capture as much semantics as possible: *"Knowledge Oriented Design"* (KOD) by Claude Vogel [22], *"Knowledge Analysis and Design Support"* (KADS) from Schreiber, Wielinga, and Breuker [23], and more recently the systemic method *"Méthode d'Analyse et de Structuration des (K)Connaissances"* (MASK) proposed by Jean-Louis Ermine [24]. This section presents the steps of the FVSOOMM method that employs temporal vector spaces (TFVS) to model knowledge with an object-oriented approach, together with the use of the composition operator and the (object "has-a" attribute) relationship [25]. The method in FVSOOMM describes the different steps and procedures to design an object-oriented knowledge base that implements fuzzy vector spaces. FVSOOMM describes the whole design cycle, from the capture of a user's knowledge requirements, to the implementation of the necessary objects of the decision support system that provides this knowledge. The fuzzy vectorial space (FVS) of each object is built with a bottom-up approach of the composition hierarchy: beginning with simple attributes, then the simple objects that only have attributes, and working higher upward toward composite objects that realize is-part-of relationships. The steps to design knowledge bases with temporal modeling are depicted in Figure 3, and the individual steps are described below:

- Step 1 of Figure 3 defines the boundaries of the system, describes the actors (internal and external actors) and objects involved in the system, and builds a dictionary of their attributes and relationships to be described in Step 2. There are three types of users:

The expert user (EXPU) designs the knowledge base of the system. The practitioner user (PU) is a domain expert and, with the EXPU, defines the attributes, objects and classes that constitute the knowledge base. Finally, there is the end-user (EU), who uses the knowledge base during his/her professional activity. During this first step, the necessary objects and attributes of the knowledge domains that are used or exchanged, as well as user requirements and roles, are defined. Data flow diagrams at different levels of granularity are built, in a similar fashion to the *"needs model"* of the Merise method [26].

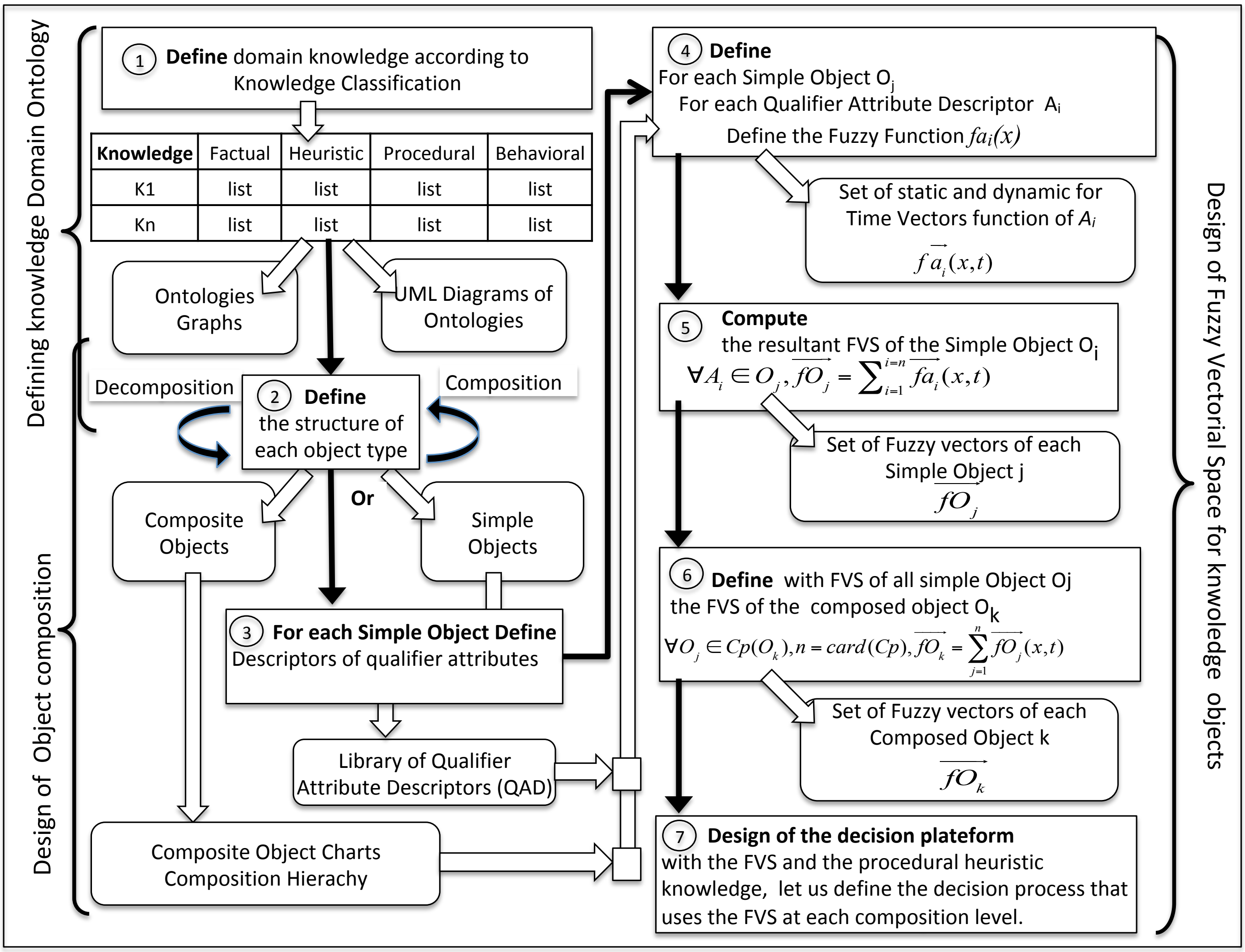

Figure 3. A method to design artificial intelligence (AI) decision support systems with fuzzy vectorial spaces.

- Step 2 of Figure 3 covers the design of the knowledge domain ontology that defines all the object classes, their composition hierarchy (is-part-of) relationships and all the relevant qualifier attributes attached to the objects of each class. Classes (object definitions) are general descriptors of the instantiated objects of the same class. A qualifier attribute (QA) is a necessary attribute for assessing the state of an instantiated object. Identifiers are specific attributes of an object that are unique, not null and are used to identify specific instantiated objects. The object-oriented model resulting from Step 2 is mainly based on the composition (is-part-of) relationships and their specific mandatory qualifier attributes [3]. Classes described in the ontology can be mapped in one or several UML class diagrams, provided that the semantic relationships of the domain ontology are well respected, as in the ontology of the diagnosis of infectious diseases [27]. However, the knowledge is described by the objects, their values, and their relationships at the internal level of the system [3]. Colloc's original metamodel of that ontology [3] was inspired from the Object Data Modeling Group (ODMG) in [28,29] and in the PhD thesis work of Ying Shen [30]. The present work proposes a new extension of Colloc's original meta-model that takes into account the fuzzy vector spaces, as depicted in Figure 4. The blue part defines the concept of type, while the red part defines class, objects, and attributes, as described by the ODMG group. The green part indicates the methods, as in Colloc's initial model. The purple part describes the TFVS extension presented earlier, qualifier attribute descriptors (QAD) that are detailed further in Figure 5, as well as the definition of fuzzy vectors that tune the properties of each object (has-a relationships) and the recursive composition of composite objects (Compose) with simple objects, using the operator $\oplus$.

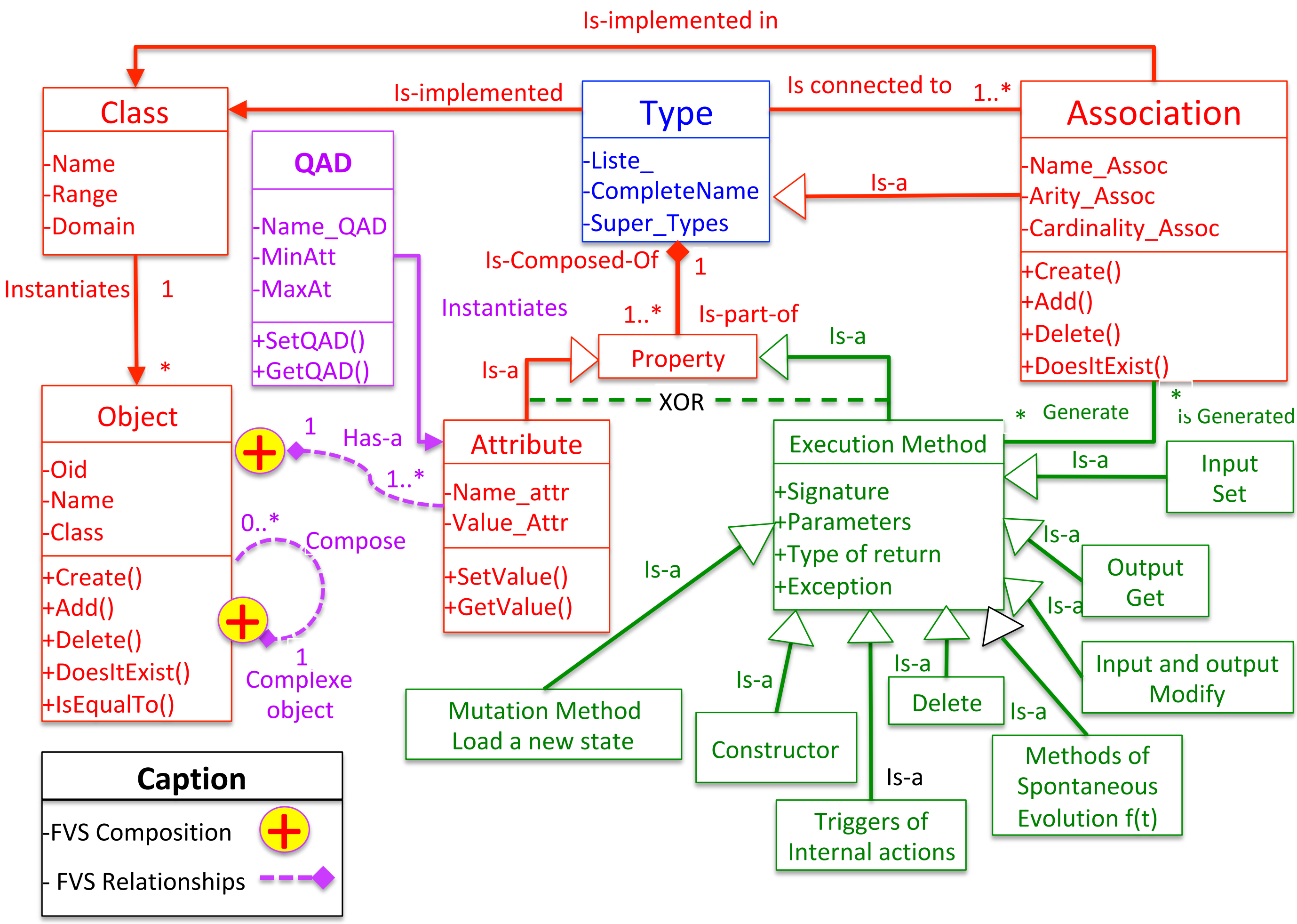

Figure 4. FVSOOMM meta-model describes the object-oriented model extended by TFVS.

- Step 3 of Figure 3 concerns the knowledge design with a fuzzy vector space. It relies on the composition relationship and the qualifier attribute descriptors (QADs) in Figure 5 to describe the characteristics of each attribute, which should be used in at least one object type descriptor (OTD) in Figure 6 to define the structure of the necessary knowledge object instances (either simple or composite). The expert (EXPU) uses the qualifier attribute descriptor interface (QADI), shown in Figure 7, to define the appropriate parameters of the QAD. Please note that Figure 7 provides a translated version of the French QADI used in this work.

- Step 4 of Figure 3 defines the fuzzy functions. For each necessary simple object O_j of the knowledge base, the expert and the practitioner use the object type descriptor interface (OTDI) to choose the QADs of each relevant attribute $O_j.a_i$ in the list of previously defined QAD (see Figure 8). Several iterations of Steps 3 and 4 are used in turn to define necessary objects (OTDs) and attributes (QADs).
- Step 5 of Figure 3 computes the resultant FVS of simple objects. For each simple object O_j, the resultant time function vector of the object is calculated from their attribute's function vectors.
- Step 6 of Figure 3 defines the FVS of composite objects. For each composite (composed) object O_k, its resultant time function vector is computed from all of its component objects O_j and attributes $O_k.a_i$.
- Step 7 of Figure 3 implements the necessary reasoning modes that use the previous objects to make a decision. The reasoning uses the vectors of the composite objects, which can recognize complex situations and make an appropriate decision. The available reasoning modes are deduction, induction, abduction, analogy, subsumption, and case-based reasoning. Case-based reasoning is described later in Section 2.3.1. Deduction can be implemented at a more macroscopic level by comparing object states (described by their resultant vectors) and taking into account time.
- The Figures 9 and 10 describe respectively the fields of the attributes and objects that are involved in the end-user interface and are respectively linked to the QAD (field idQAD) and to the OTD (field ObjTid) that define their constraints.

Fields	Description	Interface type
QADId	Attribute Qualifier Descriptor Identifier	Automatic numbering
QADW	Weight of the attribute in the object [1,100]	Input (Key-entered)
QADName	Unique descriptive name chosen by the expert	Input (Key-entered)
QADLastUpdate	Date and Hour of Last Update by the Expert	Get system date and hour when the last update by the expert
QADType	Type of value {Boolean, Character, Integer, Float, Double, String, DateHour, Enum}	Combo List to choose
QADU	Unit of the attribute if any (optional)	Input (Key-entered)
QADVmin	Minimum value of the attribute	Input (Key-entered)
QADVmax	Maximum value of the attribute	Input (Key-entered)
QADConst	The value of the attribute is constant	Radio Button () Yes () No
QADMand	The value of the attribute is mandatory (may be 0)	Radio Button () Yes () No
QADDefault	Default value when not provided by the user	Input (Key-entered)
QADlinlog	Linear value or logarithm scale (Log_{Base})	Input (Key-entered) Default is Linear
QADFuzzy	[-1,1] Fuzzy characteristic membership function Graph	Graphic of the sigmoid function
APeriodicity	Periodicity update of the attribute specified in time unit: {s,m,h,d,w,m,y} : second, minute, hour, day, week, month, year. When a revaluation will be launched (* if too short cause performance issue)	Input (Key-entered)
QADAccess	Public Attribute accessors (get, set, modify), functions that control the QAD update	Allow the expert to modify th e QAD and display it

Figure 5. Expert interface: qualifier attribute descriptor (QAD).

Fields	Description	Interface type
OTId	Object Type Identifier	Automatic numbering (+1)
OTW	Weight of the Object Type [0,100] 0 if not linked to a Complex Object	Input (Key-entered)
QTName	Name of the Object Type	Input (Key-entered)
OTDateHourUpdate	Date and Hour of Last Update of the Object Type by the Expert	Get system date and hour at each update
OTFuzzyVector	Fuzzy Value of the Fuzzy Vector Space [-1,1] That represent the state of the object. It is computed from Component object, if any, and the attributes.	Computed each time the periodicity update is elapsed
OTAttW	Sum of the Weights of the attributes of the object	Computed
OTCompW	Sum of the Weights of the object component of the object	Computed
OTNbSubObj	Number of objects types in the next level of the composition hierarchy. If the value is 0, the object type is a Simple Object type (without components)	Incremented automatically when sub-objects types are linked
OTComplexity	Integer that gives the height of the composition tree	Computed from the complexity of the heighest complex sub-object type +1
OTArrayListSubObj	ArrayList of sub-object Types that lists the composition hierarchy of the Object Type	Display the combo list of the composition hierarchy : The OTypeId and the OTypeName
OTNbQAD	Number of Qualifier Attribute Descriptors of the Object Type. If the value is 0, and if the OT is simple, the OT has a unique attribute, its identifier	Incremented automatically when QADare created and linked
OTArrayListQAD	ArrayList of QAD that lists all the Attributes Qualifier Descriptors (QAD) that defines the state of the Object Type	Display the combo list of the Qualifier Attribute Descriptors
ObjPeriodicityUpdate	Periodicity update of the object, specified in time unit: {s,m,h,d,w,m,y} : second, minute, hour, day, week, month, year. When a revaluation of the Object FuzzyVector should be launched to update the vector of the object	Input (Key-entered)
ObjTypeAccess	Public ObjTypeaccessors (get, set, modify), functions that control the Object Type update	Allow the expert to modify the Object Type and display it

Figure 6. Expert interface: object type descriptor (OTD).

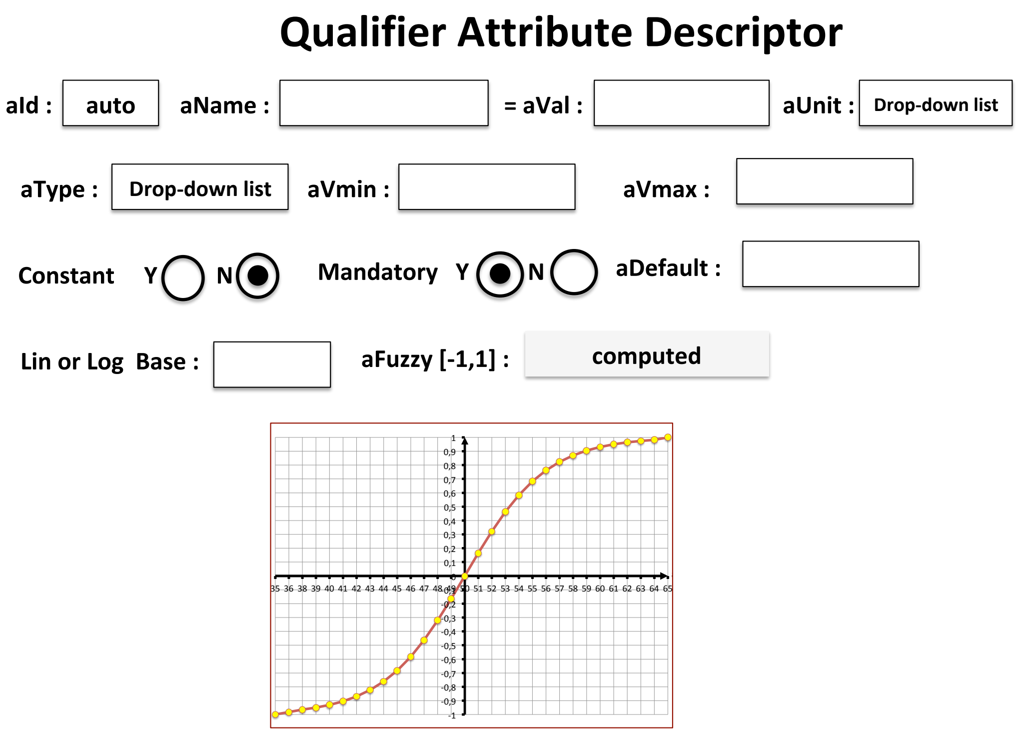

Figure 7. Expert interface: qualifier attribute descriptor interface (QADI).

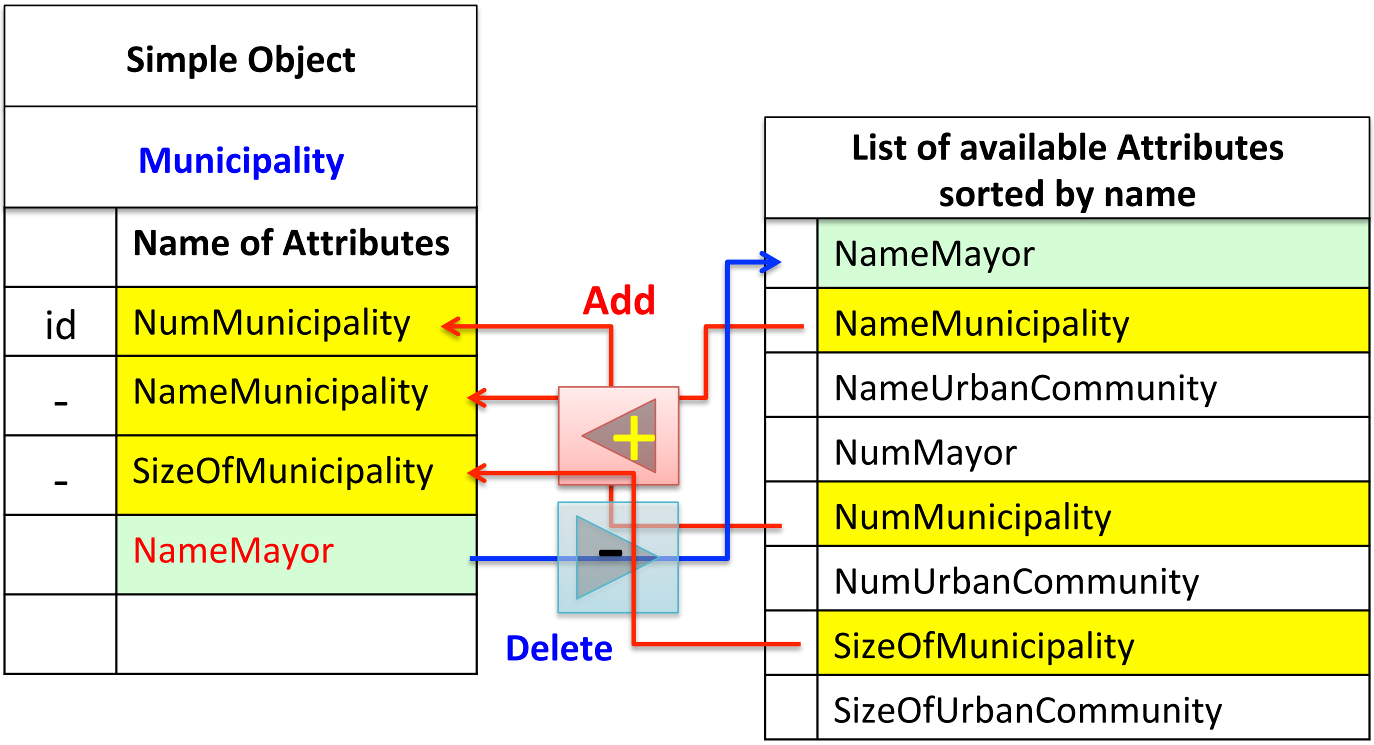

Figure 8. Object type descriptor interface (OTDI).

Fields	Description	Interface type
aId	Absolute Identifier of the attribute (independent from the object)	Automatic numbering incrementation (static **lastattid**)+1
aRel	Relative identifier of the attribute encapsulated in the object $O.a_1$ ...$O.a_n$(an attribute does not exist without being bounded to an object)	Automatic numbering incrementation (static **lastattnum**)+1
IdQAD	QAD identifier to the Qualifier Attribute Descriptor that controls the values and behavior of the attribute (integrity constraints and data type of this attribute of this object (link to the expert level)	Linked by instantiation done by the constructor of QAD
aV	Value of the attribute (should respect the constraints specified by the QAD)	Value keyed in by the end-user must respect QAD constraints
aW	Weight of the attribute in the object	Defined in the QAD of the attribute
aVFuzzy	Fuzzy value in [-1,1] is automatically computed from the value of the field aV in [min,max] at the t time defined by **aDateHourLastUpdate**	Value computed by the fuzzy membership function implemented in the QAD
aDateHourLastUpdate	Date and Hour of the Last Update of the value of the attribute and its **aVFuzzy** value	Get system date and hour when the update function is automatically launched by the system according to the periodicity of update **aPeriodicity** in the QAD
aAccess	Accessors of the attribute (get, set, modify), that control the update of the attribute of the object. They must take into account the constraints described in the QAD.	Allows the end-user to modify and update the value of the attribute according to QAD constraints

Figure 9. End-user attribute interface.

Fields	Description	Interface type
ObjId	Absolute System Identifier of the Object	Mandatory and Automatic numbering by incrementation of **LastObjId**
ObjTid	Object Type id from wich the object is instantiated	Initialized by the constructor when the object is instantiated
ObjInst	Number of this instance of Otid	Automatic numbering by the constructor give the number of Instance of Otid. The complete number is **OTid:OInst**
ObjFatherId	Object Father Id if component object, 0 if not	Id of the composite object when instantiated
ObjW	Object weight if component object, 0 if not	Weight of the object from the OTD when linked to the component object, 0 if not.
ObjFuzzyV	Object Fuzzy value in [-1,1] is automatically computed from the value of the component objects vectors and their attributes	Value computed from the vectors of component objects and attributes
ObjAttW	Sum of Weight of the attributes in the Object	Computed
ObjDateHourLastUpdate	Date and Hour of the Last Update of the value of the **aVFuzzy** value	Get system date and hour when the update function is automatically launched by the system according to the period of update defined by the variable **ObjPeriodicityUpdate** in the Object Type descriptor
ObjNbSubObj	Number of component sub-objects	Automatic incrementation when a sub-object component is created
ObjArrayListSubObj	Arraylist of identifier of component sub-objects of this object	ComboList of components ObjTid:ObjInst and their name.
ObjNbAtt	Number of attribute of this object	Automatic incrementation when the attribute is created
ObjArrayListAtt	Arraylist of identifiers of attributes aid and their QAD.	Combolist of attributes QADid:aid and QADName
ObjAccess	Accessors of the Object (get, set, modify), that control the update of the object.They must take into account the constraints described in the relevant object type	Allows the end-user to modify and update the value of the pbject according to object type constraints

Figure 10. End-user object interface.

```
// Private method of an object that computes resultant FVS of its
     attributes
Public Double GetFVSAttribValue() // return [−1,1] value of FVS
{
double  res=0; // resultant value of the vector
double SomValPos=0, SomValNeg=0, SomWPos=0, SomWNeg=0,SomWTot=0;
    // init
ArrayList latt = ObjArrayListAtt; // reference the Arraylist of
    attributes
for (i=1; i<=ObjNbAtt; i++) // for each attribute ai
  { double v=latt.get(i).avFuzzy; // read fuzzy value attribute
    ai
     double w= (double) (latt.get(i)).aW; // read weight attribute
     ai
           if(v >=0) // if the value is positive or 0
            {
     SomValPos += (w∗v∗v); // sumpositive is incremented weight.
    square(ai)
     SomWPos += w; // weight is added to sumWpositive
            }
     else
            {
     SomValNeg += (w∗v∗v); // sumnegative is incremented weight.
    square(ai)
               SomWNeg += w; // weight is added to sumWnegative
            }
     } // endfor
  SomWTot=SomWPos+SomWNeg;     // Sum of the attribute weights
```

```
   if(SomValNeg > SomValPlus) // negative sum greater than
   positive sum?
      res= - SquareRoot(-(SomValNeg-SomValPos)/SomWTot));
   else
      res= SquareRoot((SomValNeg+SomValPos)/SomWTot);
 this.ObjFuzzyV =res; // stores current value of ObjFuzzyV in the
    object
 this.ObjAttW=SomWTot; // stores current weight of attributes in
   the object
 return res;   // return the value of the attribute resultant
   vector
} // end of GetFVSAttributeValue
```

```
// Method that computes the fuzzy vector of an object  composite
   or simple
Public Double ObjectFuzzyVector()
{ double res=0;        // result of FVS vector
double somValPos=0, somValNeg=0, somWPos=0, somWNeg=0,
               attW=0; somComp=0, somTotale=0, somW=0;
// Computes the attributes FVS of this object
double resVatt=this.GetFVSAttribValue();  // Compute attributes
   FVS
double attW=this.ObjAttW;    // done by  the previous call
double n=this.ObjNbSubObj();   // Number of sub-objects of this
   object
If (n >0) { // if this is a composite object
  for (i=1; i<=n; i++) // for each sub-object
  {
     EUobject subO=ObjArrayListSubObj.get(i);   // subobject
   instance
     double v=subO.getObjFuzzyV(); //read current fuzzy sub-
   object value
     double w= (double) subO.getObjW();  // read sub-object
   weight
    if(v >=0)   {  // value is  positive
    somValPos += (w*v*v);
    somWPos += w; }                  // increments positive weights
    else   { // value is  negative
    somValNeg += (w*v*v);    // store in negative values
    somWNeg += w; }                // increments negative weights
  } // end for
somW = somWPos + somWNeg;      // Global sum of weights
if(SomValNeg > SomValPlus)    // if negative sum greater than
   positive sum
     somComp= - Math.sqrt (-(SomValNeg - SomValPos)/SomW));
 else
     somComp= Math.sqrt  ((SomValNeg+SomValPos)/SomW);
// Computes the resultant of the composition and its attributes
int Wt =  attW+somW;  // total Weight attribute + composition
double  sWAtt=Math.sign(resVatt)*resVatt*resVatt;   // attributes
    part
 double sComp=Math.sign(somComp)*somComp*somComp; // composition
somTotale=(sWatt+sComp)/Wt;
```

```
res=Math.sign(somTotale).Math.sqrt(Mat.sign(somTotale)*somTotale)
    ;
this.ObjFuzzyV=res;  // fuzzy value composition and attributes in
    object
} // endif  n>0    composite object
else // simple object with attributes only
       res=resVatt; // value of attribute FVS only
return res; // return the FVS value of object
}  // end ObjectFuzzyVector()
```

2.3.1. The Case Based Reasoning Cycle

Experience allows the advancement and enhancement of knowledge in a specific area of science. Case-based reasoning (CBR) is a model of experience that is particularly used in medicine to answer questions such as the following: What diagnosis methods were used? What treatments were chosen? What results were observed? Was the patient satisfied? All of these make a case that can be indexed and stored in an object-oriented database. These cases can then be reused in later consultations, where the additional knowledge gained from a past case can be used to provide additional benefit to enhance the relevance and the quality of care. The CBR approach uses a distance measure of the present case to past, similar cases, to retrieve cases from the database. It applies the most similar past cases to the new problem case in order to apply the most appropriate methods to solve the new problem [31–33].

Several machine learning models are used to apply the CBR approach in different knowledge domains:

- Object-oriented approaches were used to implement the CBR approach [34–36].
- The fuzzy-object relational system aims to model imprecise requirements with fuzzy objects implemented in relational databases [37]. The literature has described the comparison problem and the recursivity of assessing the relevant attributes in the objects to be compared with a distance [38].
- In health care, CBR was used to implement clinical reasoning and case modeling [39–42].
- Fuzzy logic, or stochastic approaches, were proposed to account for a set of attributes of various types from many objects, regardless of their structure. The problem becomes complex when it comes to dealing with a large number of objects [43,44].
- A Bayesian network, combined with a dynamic object-oriented approach, was used to design the resilience of engineering systems, the occurrence of accidents, and the needs of time reasoning [45,46].

CBR was implemented in algorithms of structural similarity based on object composition [47], statistical learning [48], and also digital approaches from neural networks and fuzzy logic. A CBR cycle, in a FVS extension to the work in [49], is shown in Figure 11, which illustrates how distances are used to implement the different stages of a CBR cycle. The extension to the CBR cycle provided an object-oriented CBR approach that is useful in medicine and domains where the clinical approach is relevant. The work on semantic distances tends to combine symbolic and numerical approaches [27,50].

The extension of the object-oriented model with TFVS proposed in this paper offers a new mechanism to build temporal representations of distances (time distances) between patient evolutions. This allows their comparison during an interval of time, and so provides a mechanism to find similarities of events, as shown previously in Figure 12.

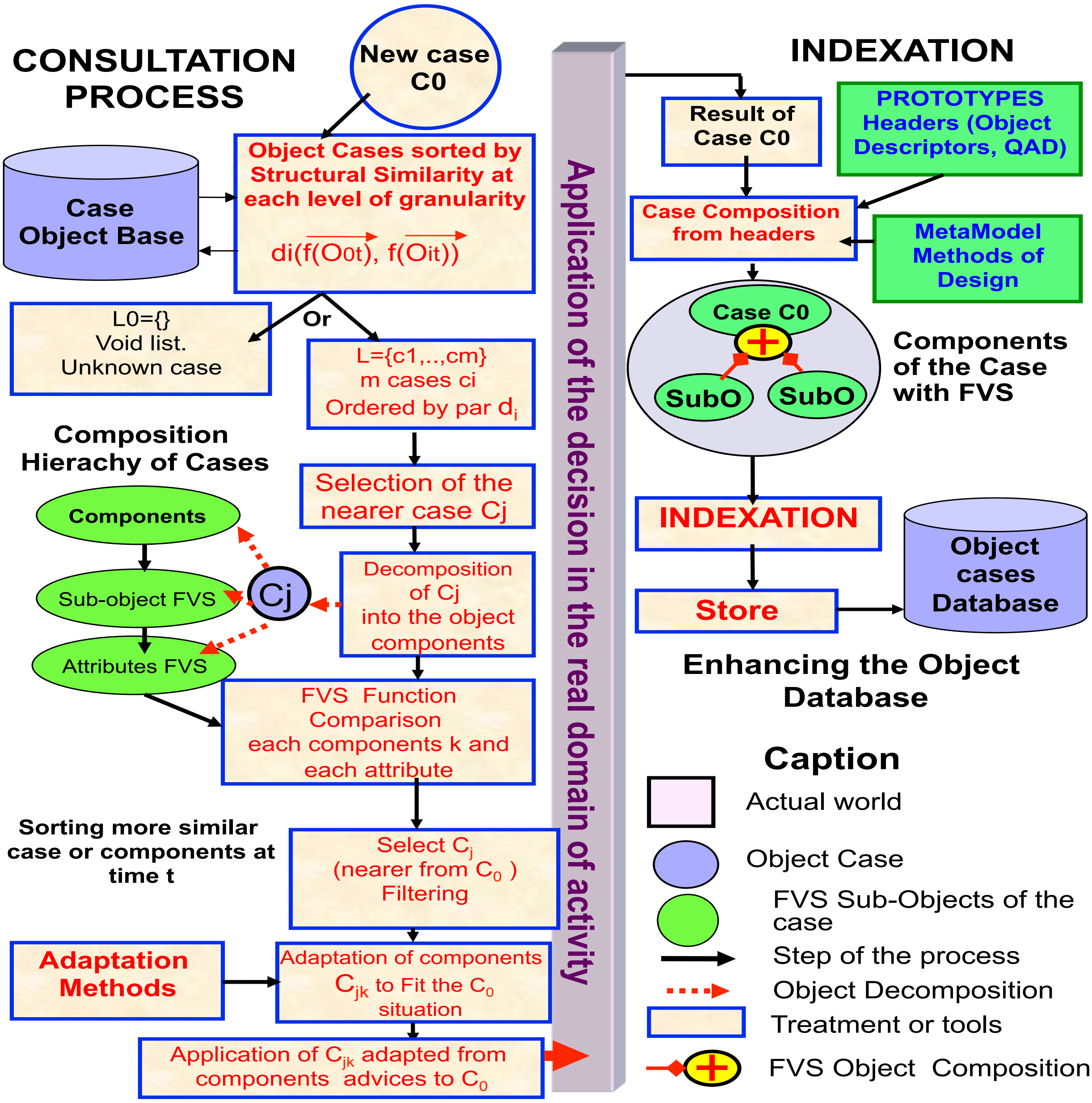

Figure 11. A case-based reasoning cycle extended from [51].

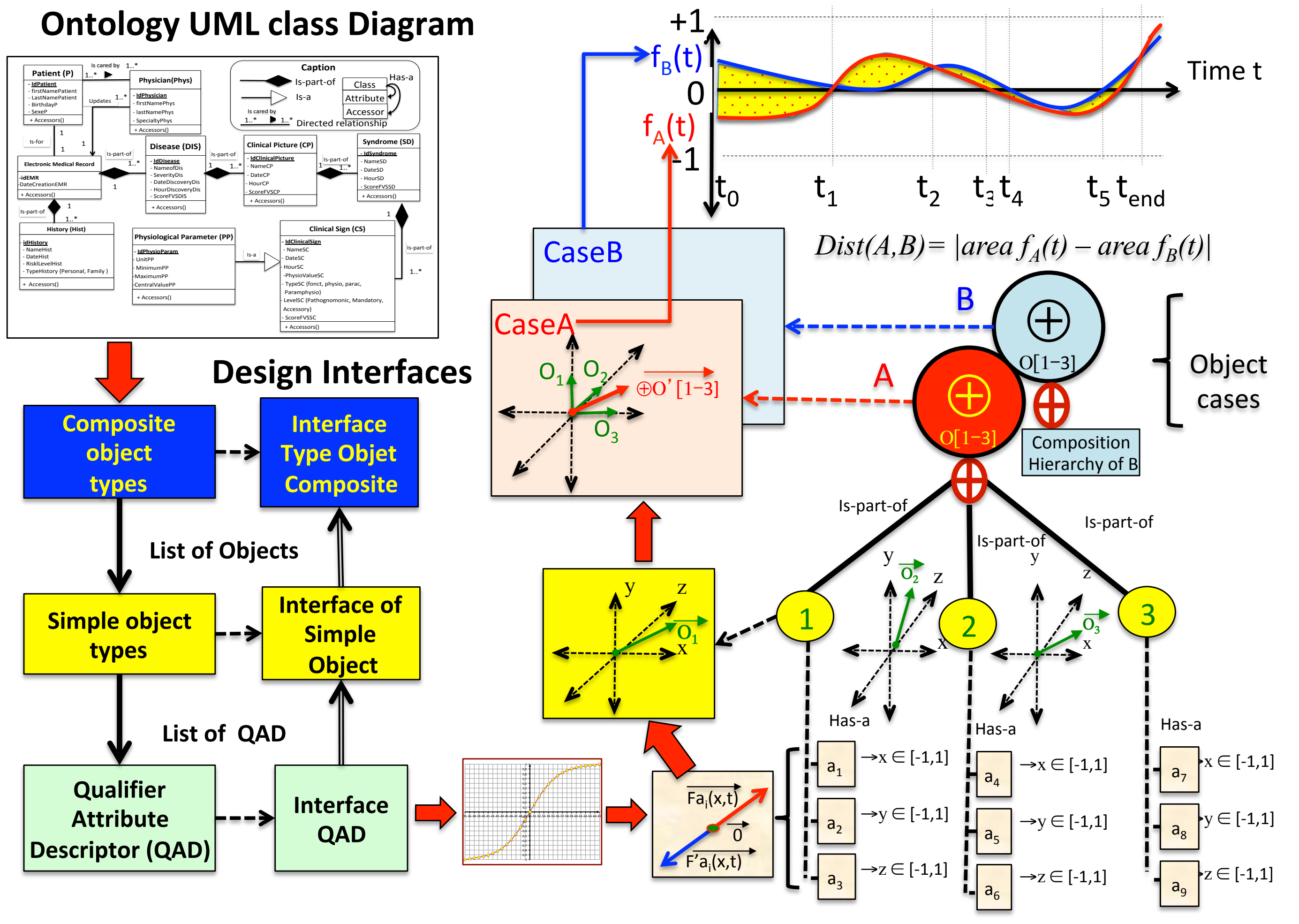

Figure 12. FVSOOMM knowledge and CBR development cycle.

2.4. Lagrange Interpolation of the FVS Function to Assess Time

Lagrange Interpolation of the FVS Function to Assess Time

$$L(X) = \sum_{j=0}^{j=n} y_j \Big(\prod_{i=0, i \neq j}^{i=n} \Big(\frac{X - x_i}{x_j - x_i} \Big) \Big). \tag{13}$$

The Lagrange interpolation for $n + 1$ points: $((x_0, f(x_0)), (x_1, f(x_1)), \ldots, (x_n, f(x_n))$ is:

$$\begin{aligned} L(X) = f(x_0) & \frac{(X - x_1)(X - x_2)(\ldots)(X - x_n)}{(x_0 - x_1)(x_0 - x_2)(\ldots)(x_0 - x_n)} \\ & + f(x_1) \frac{(X - x_0)(X - x_2)(\ldots)(X - x_n)}{(x_1 - x_0)(x_1 - x_2)(\ldots)(x_1 - x_n)} \ldots \\ & + f(x_n) \frac{(X - x_0)(X - x_1)(\ldots)(X - x_{n-1})}{(x_n - x_0)(x_n - x_1)(\ldots)(x_n - x_{n-1})} \end{aligned} \tag{14}$$

```java
package polynomlagrange;
public class PolynomLagrange {
    public static void main(String[] args) {
String spolnum="";
String spolden="";
String spoly="";
int n=4; // number of points
String [] L=new String[n];
System.out.println("Calculate polynom for "+n+" points");
for(int i=0; i<n; i++)
   {
   for(int j = 0; j<n; j++)
      {
      if (j != i)
         {
         spolnum += "(x - x"+j+")";
         spolden += "(x"+i+" - x"+j+")";
         }
      }
   L[i]="f(x"+i+")"+"."+spolnum+"/"+spolden;
   spolnum="";
   spolden="";
   }
System.out.println("L["+n+"]=");
for(int i=0; i<n-1; i++)
     System.out.println(L[i]+"+");
System.out.println(L[n-1]);
     } //end main
} //end file
/*=========== Result ===============
Calculate polynom L(X) for 4 points
L[X]=
     f(x0).(x - x1)(x - x2)(x - x3)/(x0 - x1)(x0 - x2)(x0 - x3)+
     f(x1).(x - x0)(x - x2)(x - x3)/(x1 - x0)(x1 - x2)(x1 - x3)+
     f(x2).(x - x0)(x - x1)(x - x3)/(x2 - x0)(x2 - x1)(x2 - x3)+
     f(x3).(x - x0)(x - x1)(x - x2)/(x3 - x0)(x3 - x1)(x3 - x2)
*/
```

The Lagrange interpolation Equations (13) and (14) are used to calculate a unique polynomial to assess the n-1 points $(x_{i,t}, f(x_{i,t}))$ of the FVS during an interval of time $[t_1, t_n]$ of the function $f(x_{i,t})$, that represents the evolution of the attribute x_i, or an object of the system. To compare the points of two FVS functions $fA(t)$ and $fB(t)$ on an interval of time $[t_0, t_4]$ shown on Figure 13, the polynomial $PA(t) = L(T)$ is calculated using the points $(t_0, fA(t_0), (t_1, fA(t_1), (t_2, fA(t_2), (t_3, fA(t_3)$ and polynomial $PB(t) = L(T)$, using the points $(t_0, fB(t_0), (t_1, fB(t_1), (t_2, fB(t_2), (t_3, fB(t_3)$. Then, respective areas of the polynomials are calculated by integrating these polynomials on the interval $[t_0, t_4]$, and finally, the absolute difference between the areas gives the distance. The calculation of the integral is rather simple because the primitive of a polynomial $P(x) = x^a$ is $\frac{1}{a+1}.x^{a+1}$ and the integral of a sum of monomials is the sum of the monomial's integrals. The distance is given by Equation (15).

$$distance(f_A, f_B) = \left| \int_{t_0}^{t_4} PA(t) - \int_{t_0}^{t_4} PB(t) \right| \tag{15}$$

The complexity of the algorithm to calculate the distance is $\theta(2n^2 + 2n)$, where n is the number of points, is expensive in terms of computing time. An alternative could be to compute the integral directly with the squares method. Consider two cases, case A and case B, where the comparison of the attribute x_i represented by the functions fA and fB at each time from t_0 to t_n, where we know the n measures of x_i for each of the two cases as described by Equation (16):

$$distance(fA(t), fB(t)) = \int_{t_0}^{t_n} fB(t)dt - \int_{t_0}^{t_n} fA(t)dt \simeq \sum_{i=0}^{i=n} \frac{|fB(t_i) - fA(t_i)|(t_n - t_0)}{n} \tag{16}$$

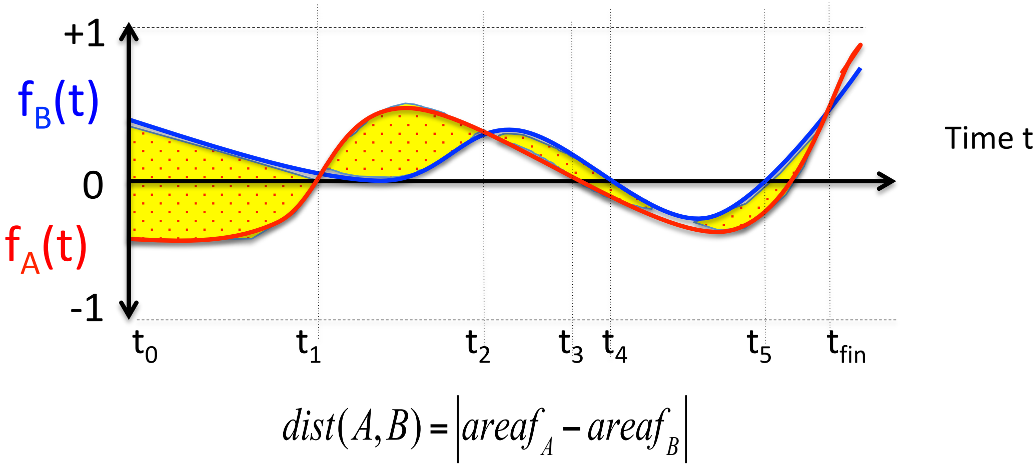

Figure 13. CBR: Distance is the difference of areas of f_A and f_B.

The calculation of this distance is much less costly θn to compute, and the precision should be sufficient to assess the similarity between the FVS of two attributes, or two objects. Distance is a key concept of the temporal CBR approach. It is used intensively to sort and index the cases along the composition tree. It is used to find what attributes and object cases show a similar evolution. The smaller the distance, the more similar the evolution of the attributes and objects. As shown in the CBR cycle of Figure 11, attributes and objects are indexed by their similarity and level of granularity, and then stored in the cases database.

2.5. Applications

This section describes two different applications designed with FVSOOMM. The first one describes a medical diagnosis of a pulmonary embolism and the follow-up after the diagnosis.

2.5.1. Decision Support System for Pulmonary Embolism Diagnosis and Follow-Up

The modeling of pulmonary embolism was done during the PhD thesis of R.A Yameogo [52]. A FVSOOMM software platform implementation for this application was developed and tested. R.A. Yameogo is a cardiologist and served as the domain expert. The initial work on the design was done in [53].

Clinical Context

Pulmonary embolism (PE) is the most severe manifestation of venous thromboembolic disease. It is defined as the obstruction of the lung arteries, or their branches, by embolisms. This is mainly caused from deep vein thrombosis (DVT) when blood clots, formed in the lower limbs, travel to the heart. The main objective of this application was to construct a clinical prediction indicator of PE with FVSOOMM. The objective of developing this indicator, which is a temporal vector space, was to improve the clinical diagnostic capabilities of PE by considering uncertainties, inaccuracies, and clinical disease progression, in order to improve the recognition and management of PE before confirmation by computed tomographic pulmonary angiography (CTPA). The advantage of using FVSOOMM is that it relies on expert knowledge and the European Society of Cardiology (ESC) [54] recommendations on the management of PE that have been validated by learned societies and experts in the field.

Graphics Summary of the PE Application

A graphical summary, depicting the different steps of the method used to develop the PE application, is illustrated in Figure 14.

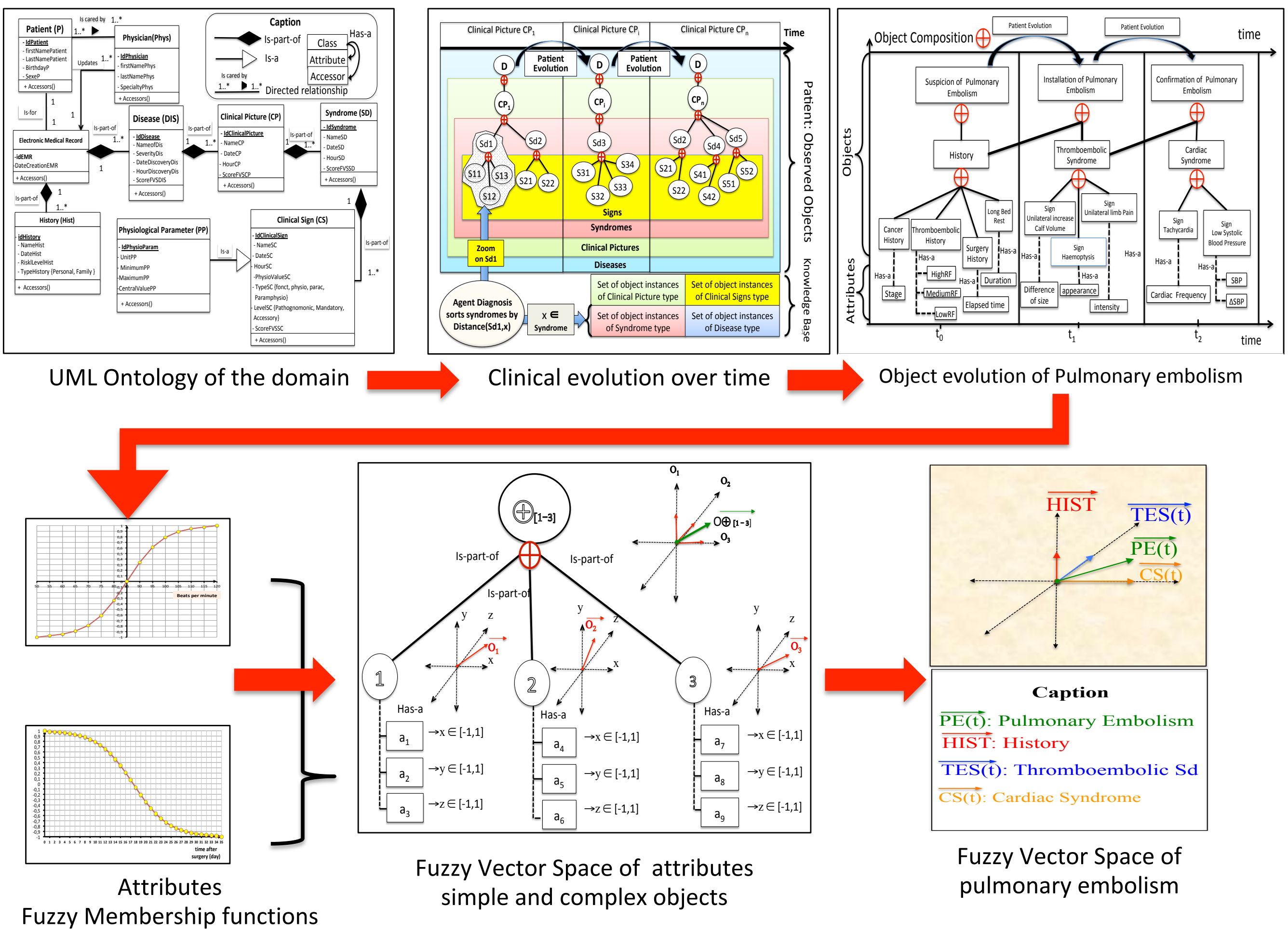

Figure 14. The graphical summary of pulmonary embolism modeling with FVSOOMM.

Variable Selection

The analysis of the PE management recommendations and scores allowed the selection and classification of variables. Three types of variables were distinguished in the diagnosis of PE: static variables and medical history that are quasi-permanent or permanent; evolutionary variables with low variability; and dynamic variables with a high and quick variation. They are reproduced, with kind permission from R.A. Yameogo's thesis [52], in Table 2.

Table 2. Types of variables used in the FVSOOM model for PE.

Variable Type	Variable Names
Static variables	Age Thromboembolic Predisposed factors Cancer Surgery or long-time bedrest
Evolutionary variables	Hemoptysis Unilateral lower limb pain Unilateral increase in calf volume
Dynamic variables	Heart rate Systolic blood pressure Systolic blood pressure variability

The UML schema of disease progression is shown in Figures 15 and 16, which define the clinical course of a typical pulmonary embolism. There is information on the following:

- Patient: age is the parameter used in the modelling of a risk probability score;
- History: thromboembolic risk factors, cancer and hospitalization history;
- Clinical symptoms and parameters: hemoptysis, unilateral leg pain, increased calf volume, heart rate and systolic blood pressure;
- Syndromes: phlebitis syndrome consisting of unilateral lower limb pain and increase in lower limb volume, and hypotension syndrome consisting of systolic blood pressure $<$90 mmHg, and/or blood pressure variation $>$40 mmHg over 15 min. The presence of one of the two syndromes is sufficient to consider the presence of a pulmonary embolism until confirmation or invalidation by CTPA.

In the case of pulmonary embolism, there are no pathognomonic or obligatory clinical signs, but there are suggestive and accessory signs that make diagnosis difficult. Moreover, it is a pathology that is very progressive in the short term, hence the interest of taking time into account in the development of a risk probability score.

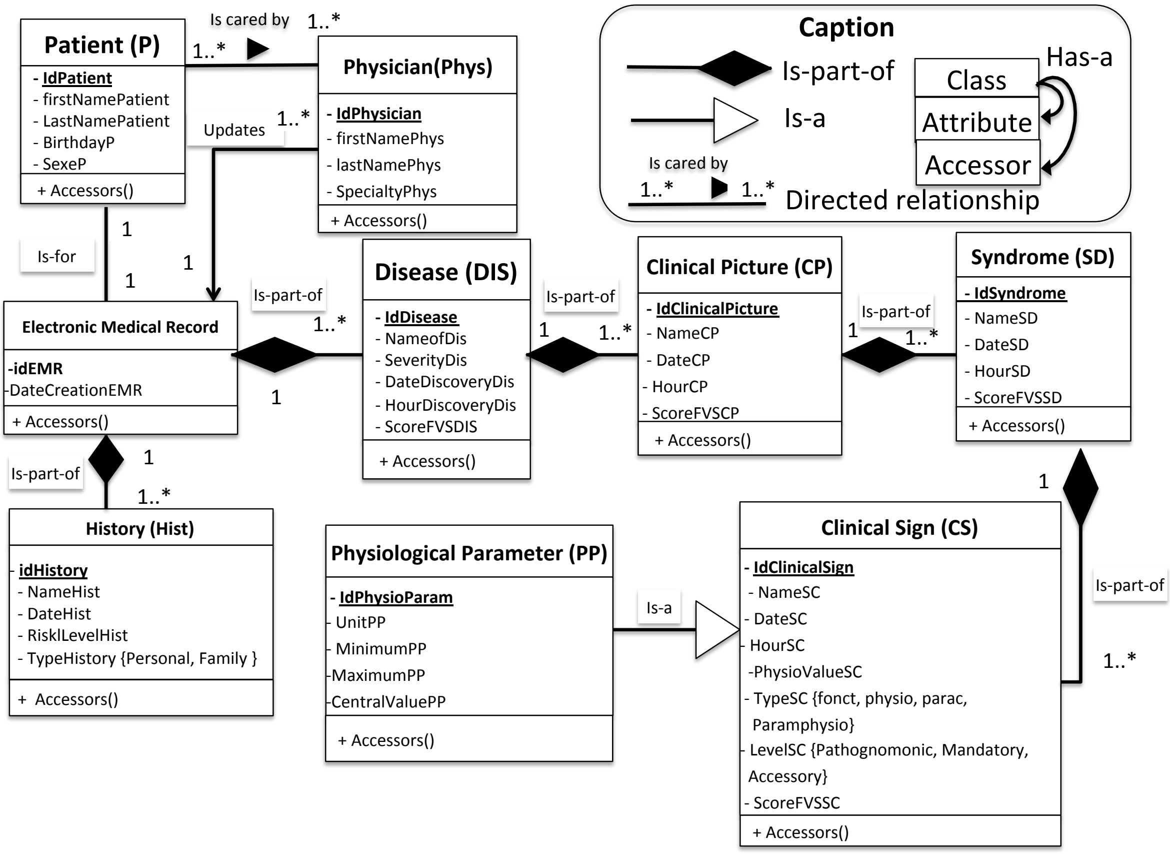

Figure 15. The UML disease diagram.

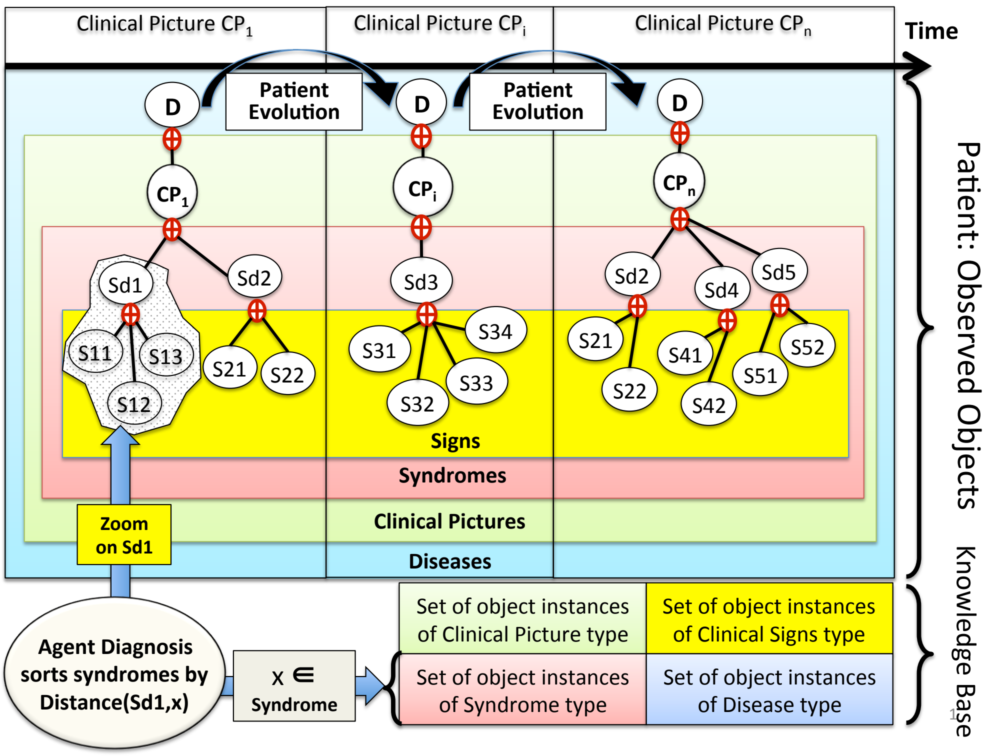

Figure 16. Clinical picture evolution object-oriented model.

Static Variables

-Age

Age is a predisposing factor in the occurrence of pulmonary embolism. The risk of PE is high when the age of a patient is above 65, moderate for an age between 50 and 65, and low if the age is below 50. The proposed approach in this paper overcomes these thresholds by using a continuous characteristic function when the attribute considered is continuous, which is the case for age (Table 1 and Figure 17).

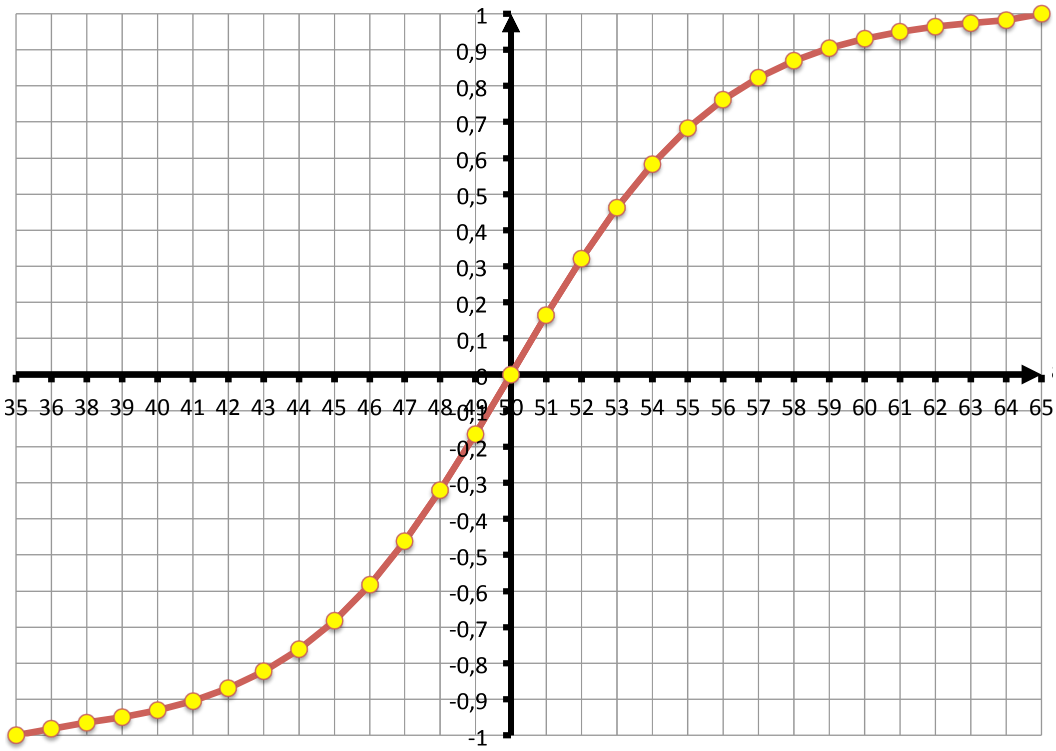

Figure 17. Fuzzy characteristic function μAge(H(x)).

- Thromboembolic risk factors

Three levels (low, moderate, or high) of thromboembolic risk factors are considered in factors that are either present or absent. A multiplicative exponential weight factor is associated with each level. For low risk, $w_i = 2$ is used, $w_j = 4$ for medium risk, and $w_k = 8$ for high risk. There are 9 low-risk factors $nblr = 9$, 18 moderate-risk factors $nbmr = 18$, and 8 high-risk factors $nbhr = 8$. It is possible to add new risk factors and to tune the curve according to local epidemiological data, if necessary. The equation for calculating the score variable is given by Equation (17).

$$Score = w_i \sum_{i=1}^{nblr=9} Lr_i + w_j \sum_{j=1}^{nbmr=18} Mr_j + w_k \sum_{k=1}^{nbhr=8} Hr_k \quad (17)$$

According to the previous score, the following fuzzy characteristic function of thromboembolic risk $\mu ThE(H(x)$ is computed. For example, when two high risk factors are found in the patient, the risk is 16 and $\mu ThE(H(x)) = 0.939$ on Figure 18.

- **Cancer risk factors**

Cancer is classified according to the stage classification of cancers:

- Stage 0—it is a carcinoma in situ or a precancerous change;
- Stage 1—the tumor is usually small and has not grown outside the organ in which it originated;
- Stages 2 and 3—the tumor is large or has grown outside the organ in which it originated into the surrounding tissue;
- Stage 4—the cancer has spread through the blood or lymphatic system to a distant location (metastatic spread).
- Stage (-1)—this is added to signify that there is no cancer.

Based on this classification, stages 0 and 1 are considered low risk, stages 2 and 3 are considered moderate risk, and stage 4 is considered high risk for VTE. Stage -1 denotes the absence of a known cancer. This stage attribute x is discrete, and Table 3 shows the values. $H(e^x)$ for cancer risk is computed in a similar way as the previous thromboembolic risk, shown in Figure 18. Some attributes are discrete and exponential like the cancer risk function described on Figure 19.

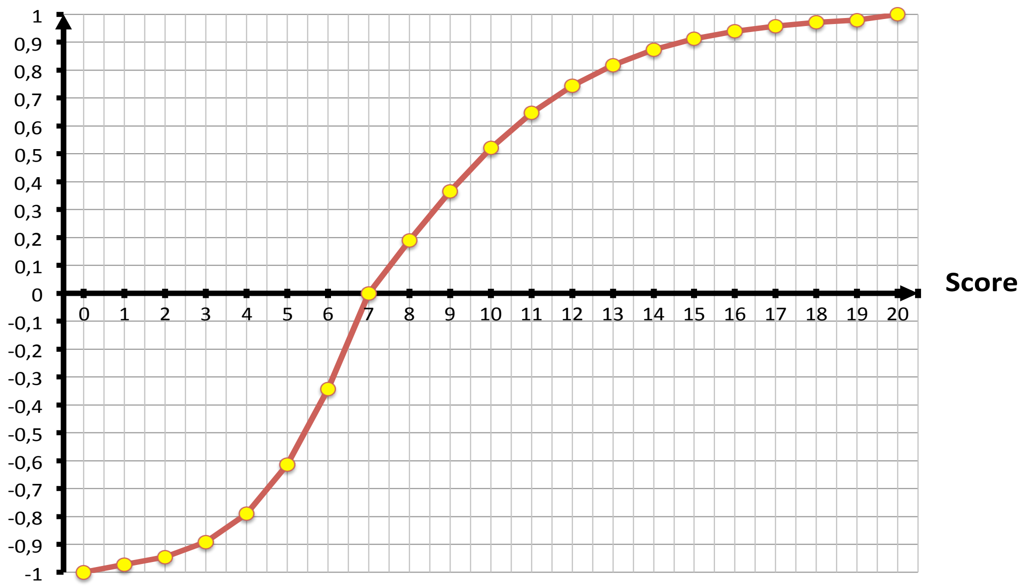

Figure 18. Fuzzy characteristic thromboembolic risk function $\mu ThE(H(x))$.

Table 3. Fuzzy characteristic function table for $\mu Cancer(H(e^x))$ risk of PE.

Stage $x \in [-1, 4]$ $(-1\ none)$	e^x	$H(e^x)$	$\mu Cancer(H(e^x))$
-1	-	-	-1
0	1.00	-0.71	-0.8
1	2.72	-0.22	-0.4
2	7.39	0.08	0.18
3	20.09	0.32	0.67
4	54.60	1.00	1

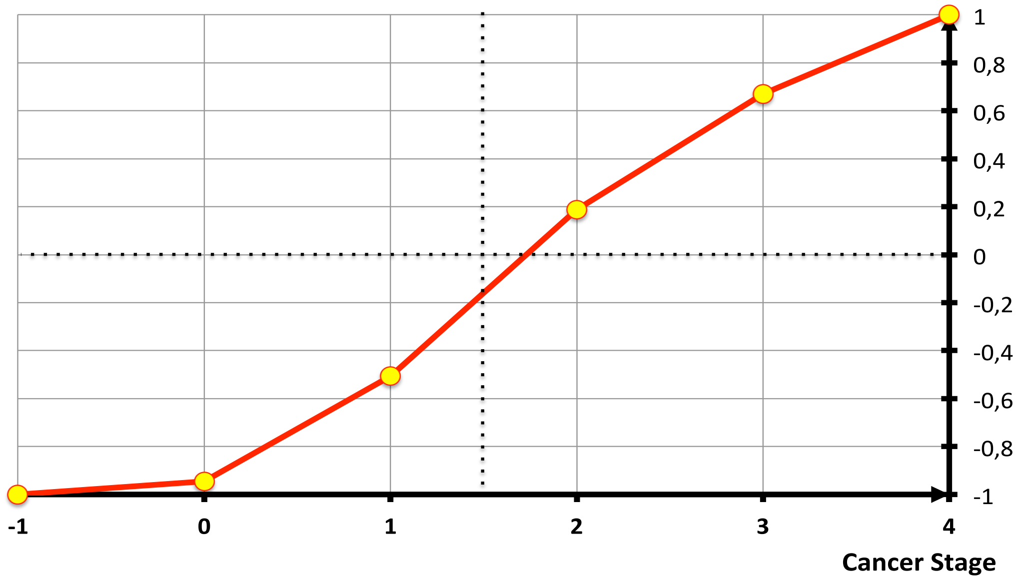

Figure 19. Fuzzy characteristic cancer risk function $\mu Cancer(H(e^x)$ of PE.

- **Hospitalization or immobilization after surgery**

A length of time of one month since surgery, immobilization after a fracture, and hospitalization, are all considered risk factors for pulmonary embolism. Since elapsed time is a continuous variable, the characteristic function $\mu RiskAfterSurgery(H(x))$ shows that the characteristic function decreases with the time (t is expressed in days) elapsed since the date of fracture or surgery(Figure 20).

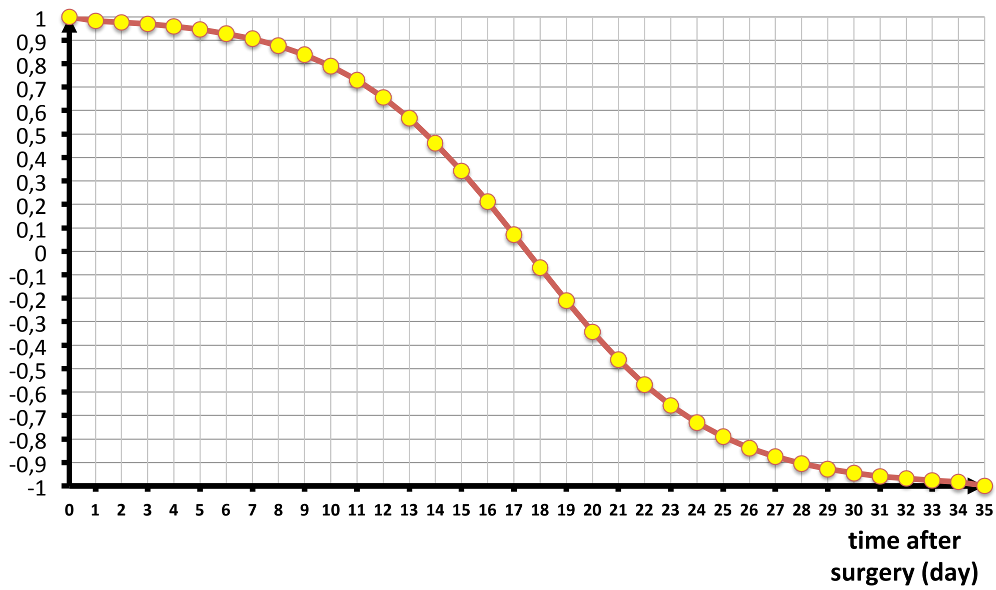

Figure 20. Fuzzy characteristic function $\mu RiskAfterSurgery(H(x)$ of PE.

Evolutionary Variables

- **Hemoptysis**

Hemoptysis is defined by its presence or absence. When present, risk is high if the hemoptysis was frank, moderate if it was blood-striped sputum, and none in the absence of hemoptysis. Hemoptysis is a discrete attribute, so the fuzzy characteristic function $\mu Hemoptysis(x)$ is discrete and expressed by Equation (18)

$$\mu RHemoptysis(t) = \begin{cases} \text{none:} & -1, \\ \text{striped sputum:} & 0, \\ \text{Frank hemoptysis:} & 1, \end{cases} \tag{18}$$

- **Unilateral pain in the lower limb**

Unilateral pain in the lower limb is defined by its presence or absence. Risk is high if the pain occurs on deep palpation of the calf, moderate if the pain is spontaneous, and low if the pain is absent. This attribute is discrete, so the characteristic function is expressed in the same way as hemoptysis by Equation (19)

$$\mu RPainLimb(t) = \begin{cases} \text{absent:} & -1, \\ \text{spontaneous:} & 0, \\ \text{palpation:} & 1, \end{cases} \tag{19}$$

- **Difference of calf size**

Unilateral enlargement of the lower limb is defined by a difference of three centimeters between the two limbs at the calf level. Risk is high if the lower limb is enlarged, moderate if the difference is between one and two centimeters, and low if the difference is less than one centimeter. This clinical sign is represented by a difference parameter of the calf size, which is continuous and expressed in centimeters. The characteristic function $\mu \Delta CalfSize(H(x,t)$ is depicted in Figure 21.

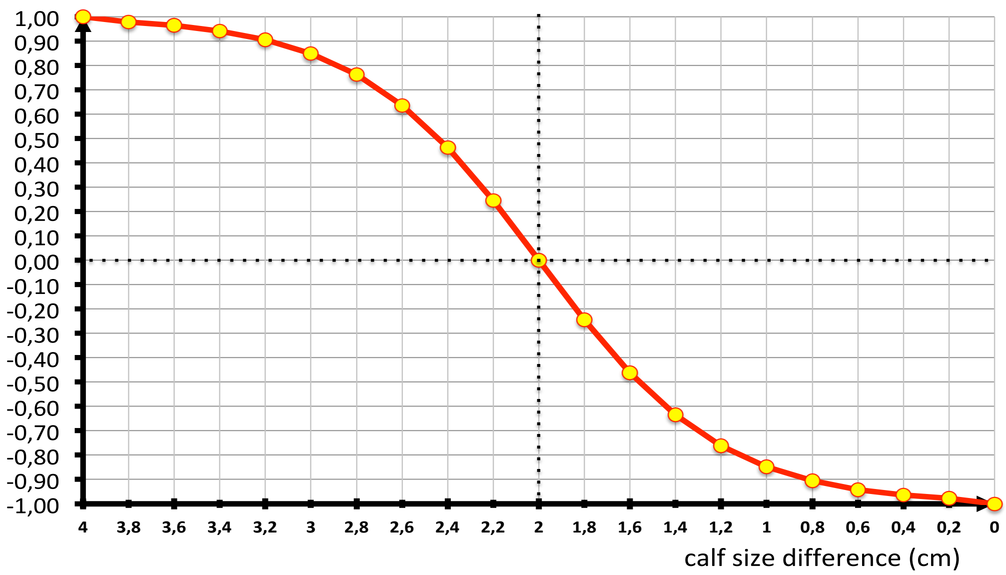

Figure 21. Fuzzy characteristic function $\mu \Delta CalfSize(H(x,t)$ risk of PE.

Dynamic Attributes

- **Heart rate**

A heart rate above 100 beats per minute indicates a high risk of VTE. The risk is moderate when the heart rate is between 75 and 100 beats per minute, and low if the heart rate is below 75 beats per minute Figure 22.

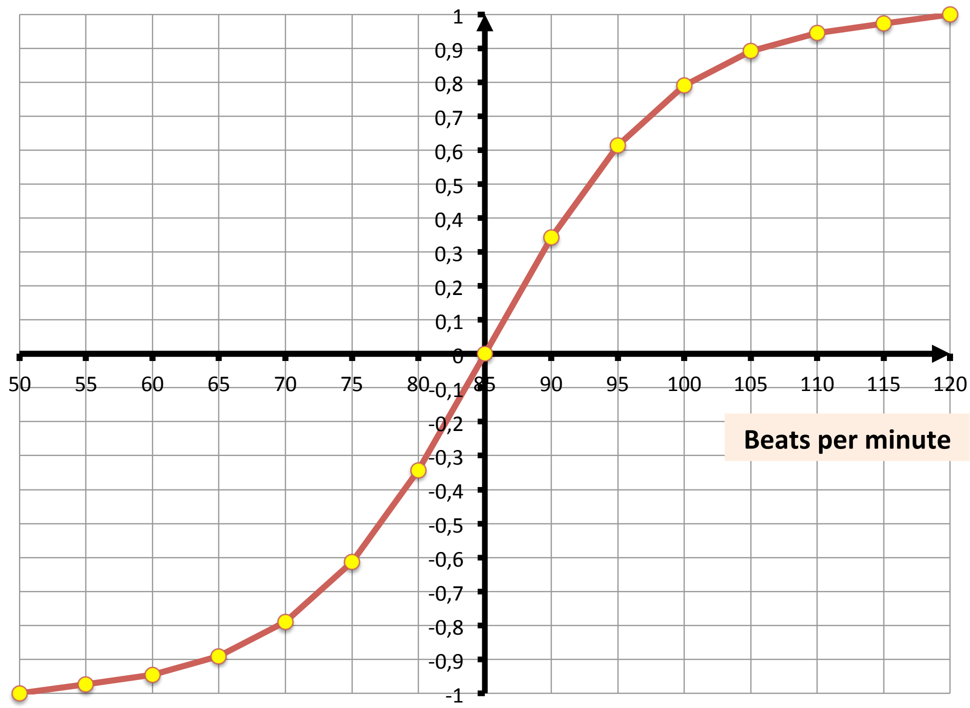

Figure 22. Fuzzy characteristic function $\mu CardiacFreq(H(x,t)$ risk of PE.

- **Low blood pressure**

When the systolic blood pressure (SBP) is less than 90 mmHg, there is hypotension, or shock, and the diagnosis of pulmonary embolism is withheld until confirmed, or negated, by computed tomographic pulmonary angiography (CTPA), if it is available. The characteristic function of low blood pressure, $\mu LBP(H(x,t)$, is presented in Figure 23.

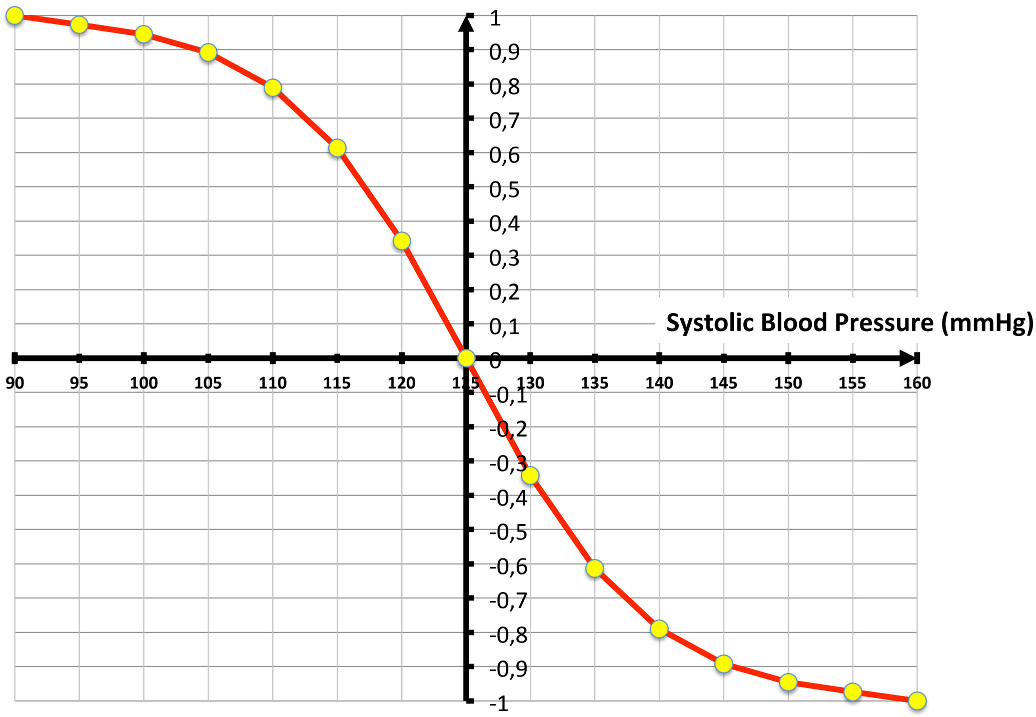

Figure 23. Fuzzy characteristic function low blood pressure $\mu LBP(H(x,t)$ risk of PE.

- **Drop of systolic blood pressure**

A variation in systolic blood pressure ($\Delta SBP = SBP_{t+1} - SBP_t$), or a drop in SBP greater than 40 mmHg for at least 15 min is considered hypotension. It has the same diagnostic value as systolic blood pressure below 90 mmHg. This is represented by the fuzzy characteristic function $\mu\Delta SBP(H(x,t)$ on Figure 24.

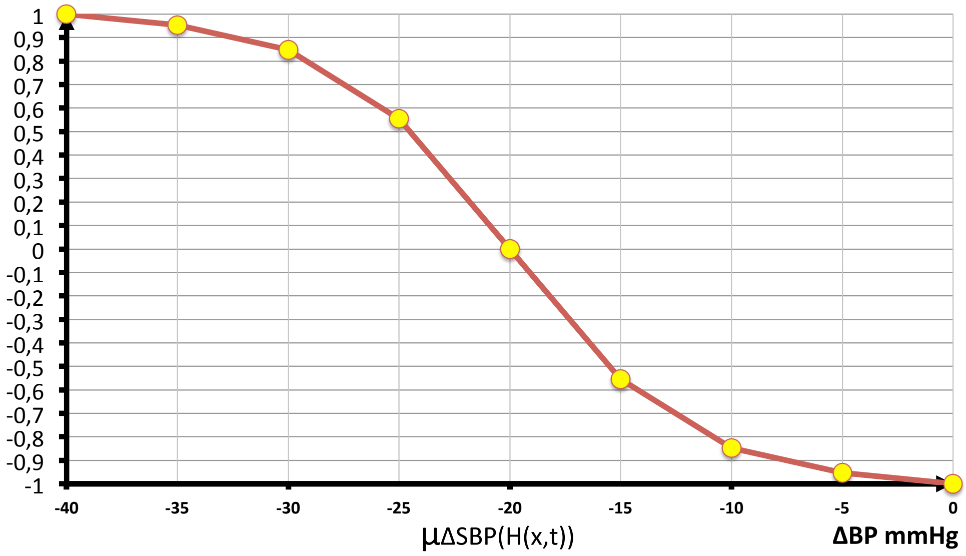

Figure 24. Fuzzy characteristic function $\mu\Delta SBP(H(x,t)$ for 15 min.

Since either criterion determines the risk associated with a drop in systolic blood pressure, we can apply the OR-inclusive operator usually used in fuzzy logic (Max), which is compatible with FVSOOMM and can be used when OR conditions are necessary on composition relationships in the UML class diagram. It is possible to combine these functions, as in Equation (20):

$$\mu RTAS(x,t_n) = Max(\mu RHypoTAS(x,t_n), \mu R\Delta tTAS(x,t_n,t_{n-15})) \tag{20}$$

A disease diagnosis results from the values of the attributes defined by successive compositions (the objects, signs, syndromes, and clinical pictures). To set up the disease, it is necessary to set up the different attributes and define the result vectors beforehand. The evaluation function for pulmonary embolism (PE) disease is between $[-1, 1]$ and is defined by the resultant of the vectors of the different variables. In this example, variables are aggregated according to the three variable categories (static, evolutionary, and dynamic). The resultant of the three vectors constitutes the PE vector and is shown in Figure 25, which illustrates the three usual diagnostic steps for the disease: step 0, history and suspicion of EP; step 1, thromboembolic syndrome; and step 2, cardiac syndrome that can be combined in a specific way in each patient.

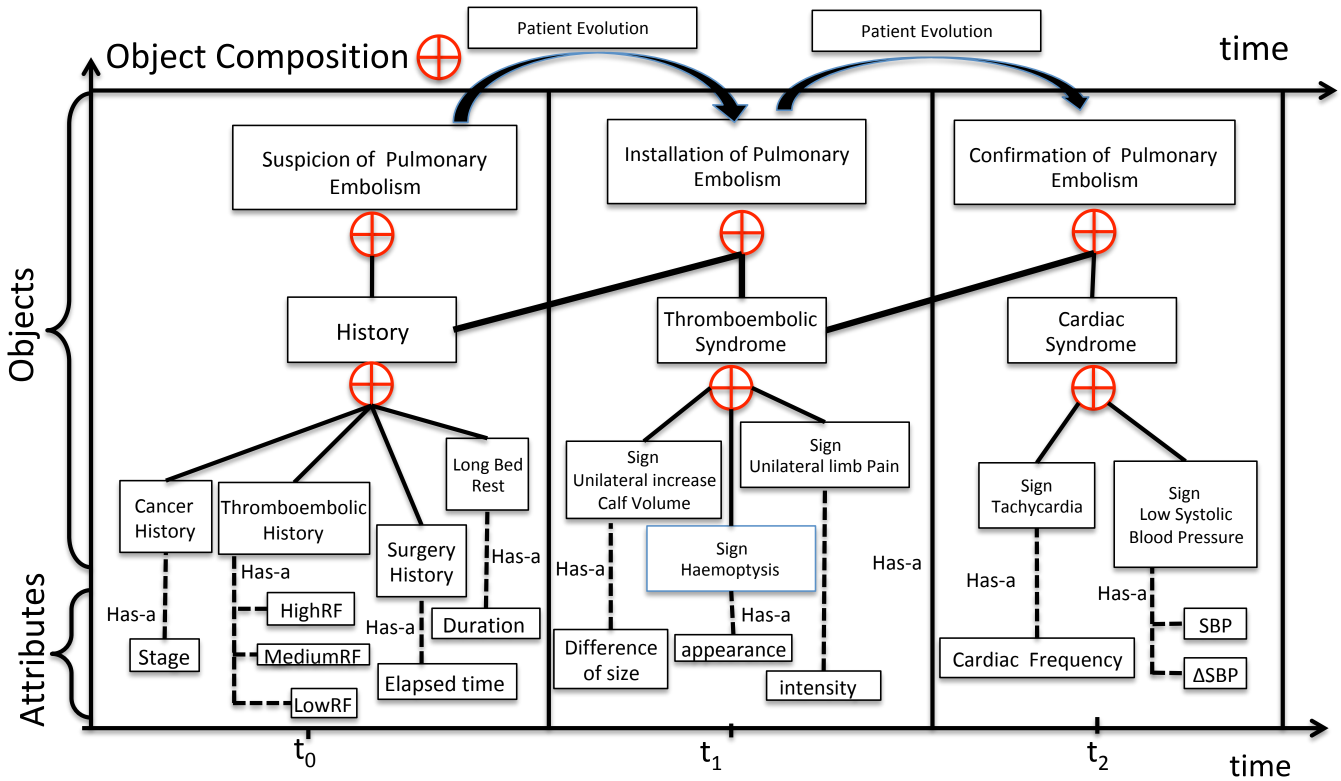

Figure 25. Pulmonary embolism's variables evolution from suspicion to diagnosis.

The corresponding FVS of pulmonary embolism $\overrightarrow{PE(t)}$ is the resultant of the vectors $\overrightarrow{HIST}$, $\overrightarrow{TES(t)}$ and $\overrightarrow{CS(t)}$, as shown on Figure 26. Note that $\overrightarrow{HIST}$ (history) is static and can just be updated in the case of missing data. As indicated from Table 2, history remains stable, while the thromboembolic syndrome and the cardiac syndrome evolve, and thus, the resultant vector $\overrightarrow{PE(t)}$, corresponding to the pulmonary embolism indicator, is changing over the time.

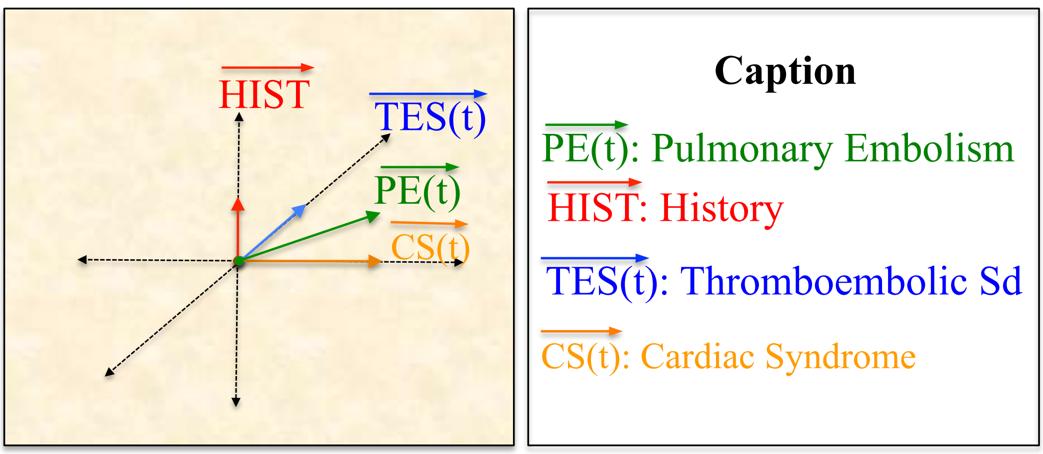

Figure 26. Pulmonary embolism's vector.

The resultant vector of pulmonary embolism $\overrightarrow{PE(t)}$ is the result of the sum of vectors expressed by Equation (21) and represented in Figure 26 [52].

$$\begin{aligned}
\overrightarrow{PE(t)} &= \overrightarrow{HIST} + \overrightarrow{TES(t)} + \overrightarrow{CS(t)};\\
\overrightarrow{HIST} &= \overrightarrow{Age} + \overrightarrow{ThE} + \overrightarrow{Cancer} + \overrightarrow{RiskAfterSurgery};\\
\overrightarrow{TES(t)} &= \overrightarrow{Haemoptysis(t)} + \overrightarrow{PainLimb(t)} + \overrightarrow{Calfsize(t)};\\
\overrightarrow{CS(t)} &= \overrightarrow{CardiacFreq(t)} + \overrightarrow{RTAS(t)}.
\end{aligned} \tag{21}$$

Thus, at each time instant, when the value of the resultant vector $\overrightarrow{PE(t)}$ tends toward 1, the risk of pulmonary embolism increases. This case [55] illustrated the importance of time modelling in clinical cases to model the steps of a diarrhea with an automaton that shows the differents diagnosis path Δ_i according to the clinical events and signs (ES), or results of investigations and the corresponding prognosis Π_i and treatments Θ_i Figure 27. The use of FVS to assess each different states of the automaton was proposed and the development of the necessary tools are presented in this paper and is illustrated with the pulmonary embolism application in the previous section.

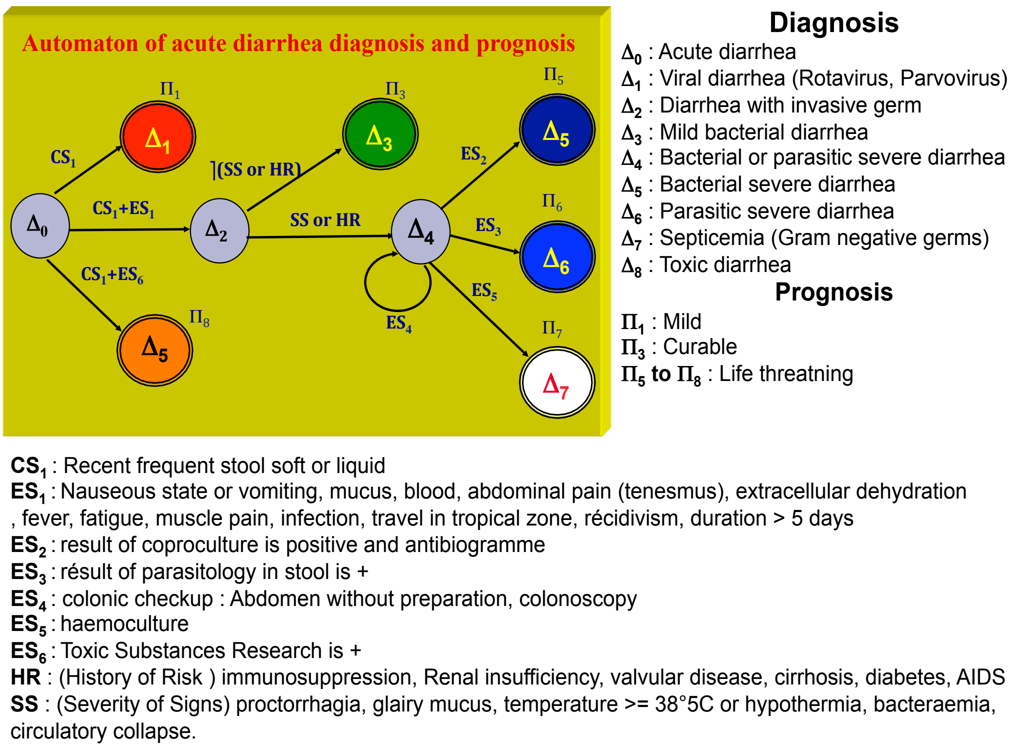

Figure 27. Diarrhea clinical automaton with diagnoses, prognosis and treatments.

2.5.2. Decision Support in Territorial Recomposition

The following example of domain knowledge describes the emotional impact of the decision maker in the territorial recomposition of French municipalities. The model describes the simulation of municipalities in regions of France and how the elected people in the government of those municipalities decide either to be involved, or not involved, in intercommunality during territorial recomposition opportunities. Lilian Loubet, through a field survey of 80 mayors in France [56], analyzed the learning process of intermunicipal cooperation [57]. This learning depends on identity, politics, and techniques. The authors believe that the emotional dimension also has an effect on the decisions of mayors with regard to psychological and emotional factors and the resulting territorial recompositions. The emotion model is the one that is used in medicine and also in management applications with the EPICE model presented in a previous paper [1]. It is based on the OCC model (Ortony, Clore and Collins) [58]. The objective of the example application presented in this paper was to the analyze a mayor's decision making. The assumptions used were that a mayor's decisions were based not only on their experiences and knowledge, but also on their psychology and emotions, especially when their municipality was one that was to be included in an intercommunality restructure that might make them surrender a part of their leadership [59]. An elected official's level of intercommunal learning affects his/her level of participation in community decision making. Conversely, the degree of participation of the elected official in the community decision (inter-municipal leadership) determines his/her level of inter-municipal learning and willingness to actively participate. This is affected according to his/her political alignment, as well as the current local, regional, and national leadership. In addition, interactions with other members (elected officials) and other communities have an influence on the decision-making process. The authors categorized several types of knowledge (factual, heuristic, and behavioral) to propose a model of the decision-making process, using a UML class diagram (shown in Figure 28) and the definition of necessary objects.

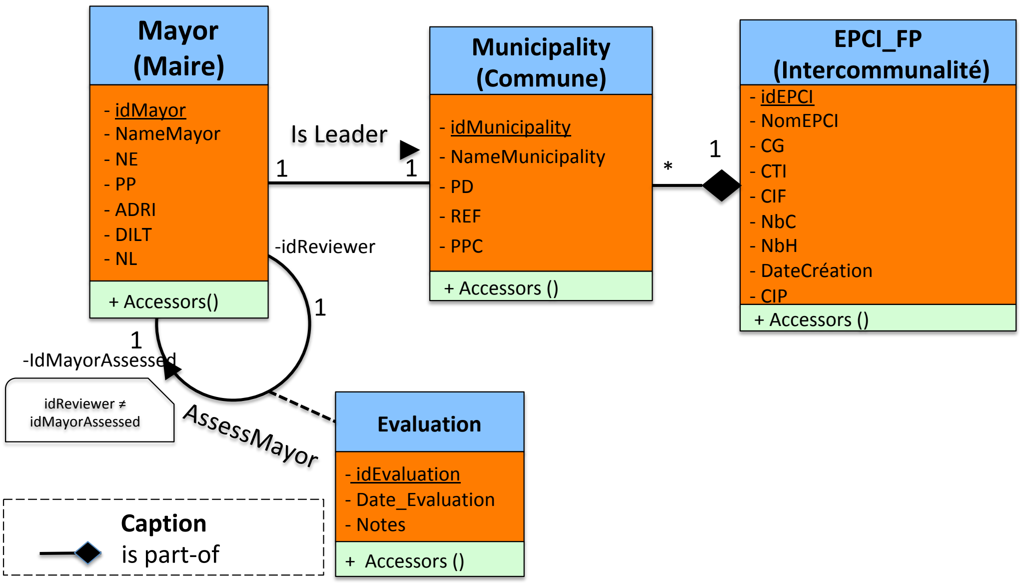

Figure 28. UML class diagram of intercommunality.

For this example, the authors modeled the emotional dimension of the mayors' decision by highlighting the cognitive variables in the spatial and temporal recomposition, using the "EPICE" model and a temporal fuzzy vector space. Three objects: "Mayor", "Municipality", and "EPCI" (Public Establishment of Intercommunity Cooperation), were retained for the construction of the decision-making system. This paper concentrates on the "Municipality" object. Other objects for this application example were described in a separate paper [59]. The choice of attributes for the different objects was based on the relationship between leadership and territory, taking into account the rational, institutional, or normative dimensions, and the dimension of identity and local cultures. As leadership is not rational, the leader is understood in his/her temporality. This temporality makes a leader a being that is in perpetual construction, where his/her decisions are moving elements. The association of the EVFT and the EPICE model allowed for a parameterization of the objects "Mayor", "Municipality", and "EPCI".

The Municipality

The attributes of the object "Municipality" are described in Table 4.

The evaluation of attributes considers the following:

- Demographic weight (PD): The demographic weight (number of inhabitants) of a municipality is materialized by the number of representatives (community councilors) it has on the community council (the deliberative body of the EPCI).
- Economic and fiscal resources (REF): This is based on different indicators. pfi is the "financial potential per inhabitant" according to the municipalities; nbpe is the level of equipment and services from the permanent equipment base (PEB). REF is a measure of the influence that a community can have within the EPCI.
- Political weight of the municipality (PPC): This is based on different indicators that were already identified during the analysis of the Mayor (M) attributes; cpp4 is the number of municipal representatives correlated to their institutional status (pp4) in the intermunicipal establishment; and M considers the mayor as the embodiment of the municipality. The results of the evaluation of the object (M) are used.

Table 4. Setting of the object "Municipality".

Object	k	Attribute	α_i	Variable	Values	Function	Value
Commune (Com)	1	PD	1	pd	$[-1;1]$	$PD = \frac{numrateur}{dnominateur}$	$[-1;1] \rightarrow [-1;1]$
	1	REF	1	pfi	$[-1;1]$	$REF = \frac{pfi + nbpe}{2}$	$[-2;2] \rightarrow [-1;1]$
			1	nbpe	$[-1;1]$		
	2	PPC	1	cpp4	$[-1;1]$	$PPC = \frac{cpp4 + 2 * M}{3}$	$[-3;3] \rightarrow [-1;1]$
			2	M	$[-1;1]$		
	$\overrightarrow{Com} = 1 * \overrightarrow{PD} + 1 * \overrightarrow{REF} + 2 * \overrightarrow{PPC}$						

For each attribute, the qualifier attribute descriptor (QAD) is defined. This provides the definition of the attribute FVS and the object FVS. The fuzzy vectors are described in Figure 29, which illustrates the fuzzy vectors of the attributes (REF, PD, PPC) and the resultant vector of the object "Municipality".

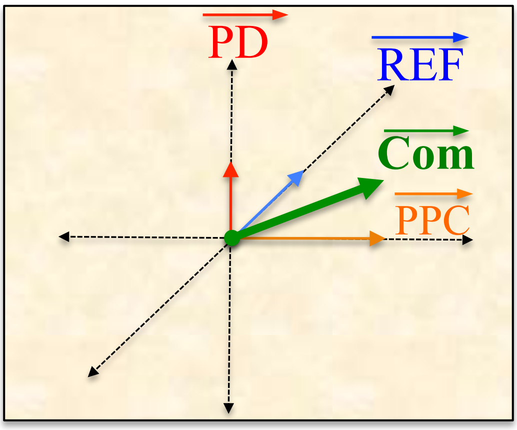

Figure 29. FVS of attributes (REF, PD, PPC) of object municipality "Commune".

The Mayor

The relationship "*the Mayor is the leader of the Municipality*" is (1,1) and thus is very similar to a composition relationship. The mayor of the municipality is described with the attributes on the UML diagram in Figure 28, where the following hold:

- NE is the level of expertise of the mayor.
- PP is the political weight of the mayor: the number of political mandates, their level (local, regional, national), and also the size of the municipality.
- ADRI is the adherence to the intercommunal framework. It assesses if a mayor is using words that seem to strongly adhere to the structuring elements of the intercommunality

(values, standards, algorithms, images). Another criterion is the number of times the name of this mayor is cited as a representative of this adherence by the other mayors involved in the intercommunality.

- DILT is the level of dilatation of the reference territoriality.
- NL is the leadership level of the mayor in intercommunality decisions, with the following values:
  - Hostile $\{-1\}$: systematically opposed to the general community movement.
  - Opponent $\{-0.75\}$: tendency to oppose the general community movement.
  - Follower $\{-0.25\}$: withdrawing from the community decision-making system, with statutory legitimacy
  - Sub-leader $\{0.25\}$: involved in decision-making spheres (associated occasionally with informal spheres), able to influence decision-making bodies, with a specific legitimacy (linked to a specific area of intervention, expertise, or political status).
  - Leader $\{1\}$: at the heart of the system, weighing in on decision-making bodies (formal and informal), able to weigh in on all areas of intervention.

The attribute values are calculated by the expert according to the weights α_i calculated from the interview of each mayor of each municipality involved in the intercommunality. The resultant vector of the mayor is calculated with Equation (22). The FVS of the mayor is shown in Figure 30. The formalism is the same as the one previously used in the pulmonary embolism example, but this time with five vectors and their respective scalars that are all normalized in $[-1, 1]$.

$$\overrightarrow{fM} = 2\overrightarrow{NE} + 2\overrightarrow{PP} + \overrightarrow{ADRI} + \overrightarrow{DILT} + 3\overrightarrow{NL} \tag{22}$$

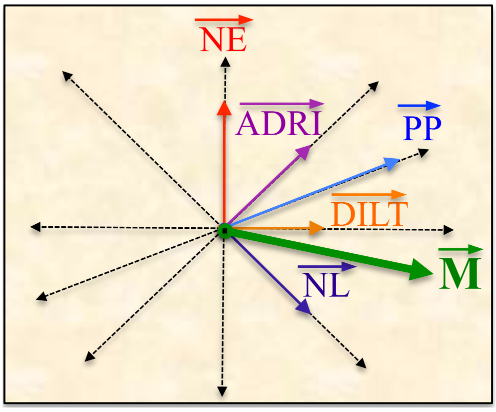

Figure 30. FVS of attributes of the object mayor, French title "Maire".

The same process is used to describe the other attributes and objects of the system according to the UML class diagram in Figure 28. The intercommunality EPCI object is also composed of municipalities "*Communities*" and an EPCI resultant vector that describes the evolution of the intercommunality, as shown by Figure 31. The municipality objects that compose the composite object EPCI are described by their vectors and should be added to $\overrightarrow{IN}$ to complete its description.

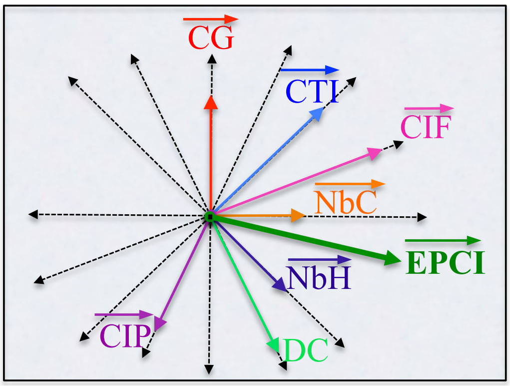

Figure 31. FVS of attributes of objects EPCI.

The intercommunality integration is assessed by the FVS using Equation (23), with the attributes listed in Table 5.

$$\overrightarrow{IN} = 2\overrightarrow{CG} + 2\overrightarrow{CTI} + 2\overrightarrow{CIF} + 1\overrightarrow{Nbc} + 1\overrightarrow{NbH} + 2\overrightarrow{DC} + 2\overrightarrow{CIP} \quad (23)$$

The knowledge was modeled without the platform, but it is currently being used in a ground study concerning the emotion of mayors in their decision to be involved or not in a territorial community [59].

Table 5. Intercommunality case attributes dictionary.

Attributes	Attribute Name
Id	Identifier of the objects (e.g., idMaire: identifier of the Mayor)
SAT_EPCI	Administrative and technical sphere of the EPCI
NE	Expertise level
PP	Political weight
ADRI	Membership of the Intercommunal Repository
DILT	Level of Dilatation of the Reference Territoriality
NL	Level of Intercommunal Leadership
PD	Demographic weight of the municipality
REF	Economic and fiscal resources of the municipality
PPC	Political weight of the municipality
EPCI FP	Public Establishment for Inter-municipal Cooperation with its own tax system
CG	Category of EPCI grouping
CTI	Number of competences transferred to the intercommunality (EPCI)
CIF	Coefficient of fiscal integration
CIP	Leadership capacity of the president of the intermunicipality
+Accessors	Public accessors (+) allow objects to be constructed, read, written and modified, and attribute values to be calculated from variables in table no. x

3. Discussion

This article follows the previous article [1] and the conference article [2]. FVSOOMM is the culmination of many previous works of the authors concerned with temporal modeling [60], ontologies [27,29,50,61], emotion modeling [62,63] and object-oriented models. The temporal-fuzzy vector space model (TFVS) is an extension of an object-oriented model based on the object composition that provides upward multiple inheritance [3,64]. This is in contrast to most object-oriented models that are mainly based on class specialization with top-down multiple inheritance, which is a Galois lattice and not a tree hierarchy as in the case of simple inheritance. This problem is often called the *"diamond problem"* (Figure 32).

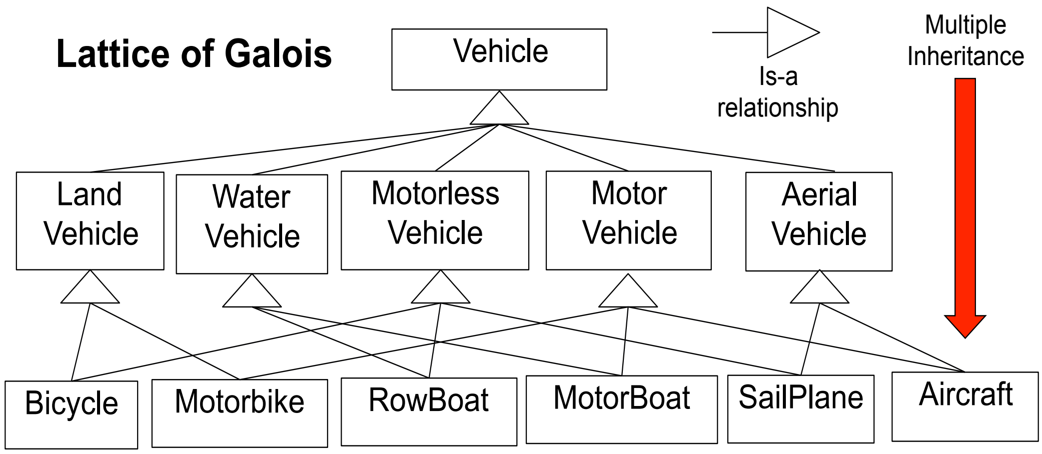

Figure 32. Multiple inheritance lattice of Galois: the vehicle example.

The lattice results in many inheritance exceptions that need to be solved manually, or with very complicated methods [65]. The composition multiple inheritance concept is similar to genetic multiple inheritance, where DNA in each cell provides bottom-up inheritance of characteristics to the cells, while epigenetic causation is top-down [66]. Some authors proposed an object-oriented approach to knowledge representation [67]. Ensing's paper provides appropriate guidelines to find objects and their relevant attributes, but does not distinguish the semantics of the relationships in the inheritance hierarchies of composition (is-part-of), generalization (is-a), or the object–attribute relationship (has-a). However, it does not address the diamond problem of class-sub-class multiple inheritance, which is the main weakness of the usual object-oriented models and languages. Software engineering also uses object-oriented approaches that try to enhance software quality and programming, such as Brad Cox [68] and Bertrand Meyer, who proposed the object-oriented language *"Eiffel"*, inspired from Pascal [69]. All of the languages that incorporate the notion of type and typed variable naturally differentiate the notion of object type from that of an object or an instance. This distinction has two advantages:

- From the conceptual point of view, there is an isomorphism between all the types of objects with the specialization relationship and all the classes of objects with the inclusion relationship. This isomorphism supports the inheritance of properties. With regard to the instantiation relationship of an object from an object type, it corresponds to a relation of an object belonging to the same class of objects. The class of objects can be assimilated to all objects having common properties that are described by a type of object. Unfortunately, the classes are not disjointed, and that leads to the diamond problem of specialization in multiple inheritance depicted in Figure 32.
- From the point of view of implementation, strong typing allows a better control of the quality of the product code. In addition, it reuses the concepts that have made structured programming successful. Adoption of the object approach in software engineering is now unanimous; encapsulation promotes the modularity, reusability, portability, security and confidentiality of software components. It facilitates the rapid prototyping of applications, using reusable object types provided in a library.

In the same way, object-oriented models are also used to develop applications with reusable models called *"design patterns"* [70,71]. However, polymorphism is often used without semantic consideration of inherited properties in a way that does not take into account the significance of the relationships used. However, this paper has shown that the semantics of the relationships (composition, generalization/specialization) must be differentiated and must not be neglected in favor of the reusability efficiency of software components, which runs the risk of system inconsistency and the loss of the user's understanding and requirements [72]. However, on the contrary, in knowledge modeling, the semantics of relationships is mandatory to appropriately design ontologies of knowledge bases in complex domains, such as infectious diseases and oncology [27,50]. The proposed approach in this paper applies fuzzy characteristic functions on object composition (is-part-of) and object–attribute relationships (has-a), whose existence is *Boolean* (that is, they either exist or do not exist) in classic object models. Thus, the proposed approach is not of fuzzy logic, but it is an extension of an object-oriented model that implements fuzzy composition and fuzzy (has-a) relationships with their qualifier attributes (QAD). This is a new extension of the ODMG meta-model in implementing fuzzy vector spaces, as shown in Figure 4. Several authors have proposed extensions of object-oriented databases with fuzzy logic [7,8], for applications such as video application databases [9]; querying on fuzzy multi-valued attributes [73]; reasoning on fuzzy object-oriented databases [74]; for deduction [75]; or to choose appropriate building elements in construction [76]. Fuzzy logic has also been proposed to assess the semantic relationships between types in ontologies [77,78] and the use of AND, or OR, operators on the specialization hierarchy [79]. The recursive process of comparing the structure of complex objects was shown by [38]. To cope with this problem, this paper's proposed approach is to compute the resultant FVS for each object in the composition hierarchy with a depth-first tree algorithm: attributes, simple objects and then composite objects. The iterative process is less costly and easier to control than a recursive process. The fuzzy object data management group (FODMG) proposed a fuzzy extension of object-oriented models to apply fuzzy logic to class specialization. They wanted to solve the multiple inheritance problem (diamond problem) by creating as many classes as necessary, with an intersection of classes created by the designer of the system (for example, a new class of teacher–student) [6,80]. This type of solution leads to implementation of as many classes as necessary, such as, student employee, retired student, etc., resulting in a proliferation of classes that could still remain incomplete. To overcome this drawback, the FVSOOMM model is based on the object composition relationship providing multiple ascending inheritance [3]. FVSOOMM proposes a new mechanism for analogic reasoning and allows temporal case-based reasoning by computing structural distances on fuzzy vectorial spaces, implemented at each level of a complex object's composition, which is a tree of components. The bottom-up multiple inheritance implementation means that no conflict is represented by the sum of the vectors, which has powerful properties of commutativity, transitivity, symmetry, and a neutral element of a vector space [1]. Moreover, the FVS scalar product allows each attribute and component to be weighted according to its importance, as given by the domain expert, to assess the object and to take appropriate decisions, for example, to perform a diagnosis, predict a suitable treatment, or to assess the prognosis of a clinical state. The FVSOOMM method was described in this paper and provides a guideline to design object-oriented knowledge bases for many different domains and temporal case databases. These contain the objects required for the domain, together with their associated attributes and the fuzzy vectors necessary to assess their current states and temporal evolutions. Until now, FVSOOMM was used manually to develop knowledge bases. This paper presents a prototype version of the new platform used to develop knowledge base objects, their attributes and fuzzy vector spaces, and to verify their consistency. The authors believe that this is a promising approach for the development of knowledge-based systems in complex areas, such as health, management, medicine, and other domains, where time is critical for decision making. The proposed model is suited to many different domains. For example, the temporal evolution of the objects in the municipal recomposition example application is much slower (semesters or years) than in the pulmonary embolism application (days, hours, or minutes). However, the proposed model is capable of describing the time parameter in these two very different applications by utilizing the object-oriented time class that is available in the Java lime package.

Future Work

Two different domains were provided as examples in the paper. The platform described in this paper is currently available with its appropriate interfaces (expert, practitioner, and end-user) to design, implement, and to use the knowledge objects and their attributes. The examples have shown that it can be used to display object and attribute states in various domains and processes. For example, to perform a diagnosis, to allow stakeholders to monitor the evolution of patients and municipalities, and to make appropriate decisions. This paper demonstrated how to implement temporal CBR within FVSOOMM. This reduces the complexity and computing time of using Lagrange interpolation to compare two fuzzy vector functions from $\theta(2.n^2 + 2n)$ to θn. This is a very important contribution because it simplifies the implementation of object-oriented temporal case-based reasoning. Future work is still required to simplify and enhance the efficiency of the distance algorithm. Additionally, CBR interfaces still need to be developed to take into account the temporal case-based reasoning features that are not currently fully implemented in the current FVSOOMM platform to compare the evolution of cases. Future work can still be done for the improvement of the current platform. It does not yet provide appropriate interfaces to zoom in on the evolution of some relevant component object of clinical events during a specific interval of time. This is future work that would allow a practitioner (e.g., physician or mayor) to discover new events and knowledge from the present case. As an example, in the treatment of a patient, consideration of what could be appropriate to enhance decisions for this situation, such as what would have enhanced the patient's clinical state and what was useless, or worse, a mistake. This kind of feature will be very interesting for assessing what was wrong and what was beneficial in the treatment of such cases, to take advantage of, and to fully appreciate and profit from, previous clinical experiences. The implementation of FVSOOMM is currently being used for pulmonary embolism diagnosis in Burkina-Faso; however, the success of this is not yet available. Similarly, in the territorial recomposition intercommunality example application, the mayor can study the impact of past decisions on the municipality evolution. The CBR interface also remains to be designed and developed in future works. In the territorial application, the cases were modeled and made available on a spreadsheet with the appropriate fuzzy functions already computed. The knowledge was designed by Lilian Loubet, who is an expert of the field. Fuzzy vectors were developed; however, the platform does not currently provide an end-user interface to municipalities and urban communities. Future work will provide that interface to make the knowledge available to mayors and stakeholders to help them make appropriate decisions that rely on the evolution of the municipalities that are inside, or outside, their neighbor's urban intercommunalities.

4. Conclusions

This work relies on a previous paper that showed the advantages of an object-oriented model, extended with fuzzy vector spaces. This paper provided further extensions to that model. Among the contributions presented in this paper, two important contributions are the TFVS model, which is a fuzzy object-oriented model that allows us to design the values of the attributes and the behavior of the objects of the system over time, and the detailed methods in FVSOOMM that describe the necessary steps to design the TFVS from a UML class diagram representing the domain ontology. These two contributions are important in computer science, as they provide a framework for temporal modeling in many domains, where fuzzy object-oriented approaches can be used and where dynamic behavior and interactions between objects must be taken into account. For example, current work is developing an application of the TFVS model on territorial policy for urban communities in France. This approach hopes to understand and explain how mayors make their decisions and indicate power issues in urban communities. TFVS can be used to represent the evolution speed, acceleration, and bifurcation of the flow of thoughts. Artificial intelligence (AI) can be useful to us if, and only if, it can preserve a scrupulous respect for the identity and freedom of end users.

5. Patents

FVSOOMM model and methods are registered in "l'agence des dépots numériques".

Author Contributions: Conceptualization, J.C.; methodology, J.C., P.S., R.A.Y.; software, J.-B.C., P.B.; validation, pulmonary embolism application, R.A.Y. and J.C.; territorialization application, L.L.; formal analysis, J.C.; investigation, R.A.Y., L.L.; resources, R.A.Y., L.L.; data curation, none; writing, J.C.; writing—review and editing, J.C., R.A.Y. and P.S.; visualization, J.C.; supervision, J.C.; project administration, J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Sample Availability: Samples of the compounds are available from the authors.

Abbreviations

The following abbreviations are used in this manuscript:

ADRI	Membership of the Intercommunal Repository
AI	Artificial Intelligence
CBR	Case-Based Reasoning
CG	Category of EPCI Grouping
CIF	Coefficient of Fiscal Integration
CIP	Leadership capacity of the President of the intermunicipality
CTI	Number of competences transferred to the intercommunality (EPCI)
CTPA	Computed Tomographic Pulmonary Angiography
DILT	Level of Dilatation of the Reference Territoriality
DSS	Decision Support System
EPCI FP	Public Establishment for Inter-municipal Cooperation with its own tax system
EU	End-User
EVF	Espace Vectoriel Flou
EXPU	Expert User
FVS	Fuzzy Vector Space
FVSOOM	Fuzzy Vector Space object-oriented Model
FVSOOMM	Fuzzy Vector Space object-oriented Model and Method
KADS	Knowledge Analysis and Design Support
KOD	Knowledge Oriented Design
MASK	"Method of Analysis and Structure of the Knowledge"
NE	Expertise level
NL	Level of Intercommunal Leadership
OTD	Object Type Descriptor
OTDI	Object Type Descriptor Interface
PD	Demographic weight of the municipality
PP	Political weight
PPC	Political weight of the municipality
PU	Practitioner User: Expert of the Domain
QA	Qualifier Attribute
QAD	Qualifier Attribute Descriptor
QADI	Qualifier Attribute Descriptor Interface
REF	Economic and fiscal resources of the municipality
TFVS	Time Fuzzy Vectorial Space
UML	Unified modeling Language

References

1. Colloc, J. Fvsoomm a Fuzzy Vectorial Space Model and Method of Personality, Cognitive Dissonance and Emotion in Decision Making. *Information* **2020**, *11*, 229. [CrossRef]
2. Colloc, J. The evolution of artificial intelligence towards autonomous systems with personality simulation. In Proceedings of the 2019 European Simulation and Modelling Conference ESM'2019, Palma de Mallorca, Spain, 28–30 October 2019; pp. 61–72.
3. Colloc, J.; Boulanger, D. Un Modèle objet pour la représentation de connaissances empiriques. In Proceedings of the Informatique Cognitive des organisations (ICO'89), Moulin, Simian, 12 June 1989; pp. 119–140.
4. Zadeh, L.A. Fuzzy Sets. *Inf. Control* **1965**, *8*, 338–353. [CrossRef]
5. Zadeh, L.A. Is there a need for fuzzy logic? *Inf. Sci.* **2008**, *178*, 2751–2779. [CrossRef]
6. Cross, V.; Caluwe, R.D.; VanGyseghem, N. A perspective from the Fuzzy Object Data Management Group (FODMG). In Proceedings of the 6th International Fuzzy Systems Conference, Barcelona, Spain, 5 July 1997. [CrossRef]
7. Yazici, A.; Koyuncu, M. Fuzzy object-oriented database modeling coupled with fuzzy logic. *Fuzzy Sets Syst.* **1997**, *89*, 1–26. [CrossRef]
8. Ma, Z.; Zhang, W.; Ma, W. Extending object-oriented databases for fuzzy information modeling. *Inf. Syst.* **2004**, *29*, 421–435. [CrossRef]
9. Ozgur, N.B.; Koyuncu, M.; Yazici, A. An intelligent fuzzy object-oriented database framework for video database applications. *Fuzzy Sets Syst.* **2009**, *160*, 2253–2274. [CrossRef]
10. Ruspini, E.H. On the Semantics of Fuzzy Logic. *Int. J. Approx. Reason.* **1991**, *5*, 45–88. [CrossRef]
11. Montiel, O.; Castillo, O.; Melin, P.; Sepulveda, R. Medaitive Fuzzy Logic for Controlling Population Size in Evolutionary Algorithms. *Intell. Inf. Manag.* **2009**, *1*, 108–119.
12. Lorestani, A.N.; Omide, M.; Shoobaki, S.B.; Borghei, A.M.; Tabatabaeefar, A. Design and Evaluation of a Fuzzy Logic Based Decision Support System for Grading of Golden Delicious Apples. *Int. J. Agric. Biol.* **2006**, *8*, 440–444.
13. El-Nasr, M.S.; Yen, J.; Ioerger, T.R. FLAME-Fuzzy Logic Adaptive Model of Emotions. *Auton. Agents Multi-Agent Syst.* **2000**, *3*, 219–257. [CrossRef]
14. Zadeh, L.A. Fuzzy algorithms. *Inf. Control* **1968**, *12*, 99–102. [CrossRef]
15. Zadeh, L.A. *Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. E.R.L. Memo M 342*; Technical Report; University of California: Berkeley, CA, USA, 1972.
16. Zadeh, L.A. The concept of linguistic variable and its application to approximate reasoning, Part 2. *Inf. Sci.* **1975**, *8*, 301–357. [CrossRef]
17. Zadeh, L.A. The concept of linguistic variable and its application to approximate reasoning Part 3. *Inf. Sci.* **1975**, *8*, 43–80. [CrossRef]
18. Katzaras, A.K.; Liu, D.B. Fuzzy Vector Spaces and Fuzzy Topological Vector Spaces. *J. Math. Anal. Appl.* **1977**, *58*, 135–146. [CrossRef]
19. Lubczonok, P. Fuzzy Vector Space. *Fuzzy Sets Syst.* **1990**, *38*, 329–343. [CrossRef]
20. Vozkoglou, M.G. A Fuzzy Model For Analogical Problem Solving. *Int. J. Fuzzy Log. Syst.* **2012**, *2*. [CrossRef]
21. Agbossou, I. Fuzzy cognitive maps-based modelling of residential mobility dynamics: GeoComputation approach. *Plurimondi* **2017**, *17*, 169–190.
22. Vogel, C. *Génie Cognitif*; Number 2-225-81382-5; Masson: Paris, France, 1988; p. 196.
23. Schreiber, G.; Wielinga, B.; Breuker, J. *KADS: A Principled Approach to Knowledge-Based System Development*; Academic Press: London, UK, 1993.
24. Ermine, J.L. Knowledge Management with the MASK method. In *Knowledge Management for Sustainable Development*; Munshi, U.M., Sharma, V.K., Eds.; Medtech: Edgartown, MA, USA, 2013.
25. Boulanger, D.; Colloc, J. Une Méthode Orientée Objet Pour L'élaboration de Systèmes Experts. 1993. Available online: https://hal.archives-ouvertes.fr/hal-02586114/document (accessed on 10 December 2021).
26. Tardieu, H.; Rochfeld, A.; Colletti, R. *La Méthode Merise*; Les Éditions d'organisation: Paris, France. 2000.
27. Shen, Y.; Yuan, K.; Chen, D.; Colloc, J.; Yang, M.; Li, Y.; Lei, K. An ontology-driven clinical decision support system (IDDAP) for infectious disease diagnosis and antibiotic prescription. *Artif. Intell. Med.* **2018**, *86*, 20–32. [CrossRef]
28. Bartels, D.; Berler, M.; Eastman, J.; Gamerman, S.; Springer, A.; Strickland, H.; Wade, D. *The Object Database Standard ODMG 2.0*; Morgan Kaufmann Publishers: Burlington, MA, USA, 1997.
29. Shen, Y.; Jacquet-Andrieu, A.; Colloc, J. Un Système Multi-Agents d'Aide à la Décision Clinique Fondé sur les Ontologies. In Proceedings of the Sixth International Workshop on Medical and Healthcare Applications (AMINA'2012), Mahdia, Tunisie, 17 January 2012; pp. 136–145.
30. Shen, Y. Elaboration d'Ontologies Médicales Pour une Approche Multi-Agent d'Aide à la Décision Clinique. Ph.D. Thesis, Université Paris Ouest Nanterre La Défense, Nanterre, France, 2015.
31. Kolodner, J.L. Maintaining Organization in a Dynamic Long-Term Memory. *Cogn. Sci.* **1983**, *7*, 243–280. [CrossRef]
32. Kolodner, J.L. *Case-Based Reasoning*; Morgan Kaufmann: Burlington, MA, USA, 1993.
33. Aamodt, A.; Plaza, E. Case-Based Reasoning, Foundational Issues, Methodological Variations, and System Approaches. *AI Commun.* **1994**, *7*, 39–59. [CrossRef]

34. Dubitzky, W.; Bell, D.; Hughes, J. A generic, object-oriented case-knowledge representation scheme, and its integration into a wider information management scenario. *Expert Syst.* **1996**, *13*, 219–233. [CrossRef]
35. Khan, A.A.; Chaudhry, I.A. Object Oriented Case Representation for CBR Application in Structural Analysis. *Appl. Artif. Intell.* **2015**, *29*, 335–352. [CrossRef]
36. Lee, J. Modeling imprecise requirements with fuzzy objects. *Inf. Sci.* **1999**, *118*, 101–119. [CrossRef]
37. Cuevas, L.; Marín, N.; Pons, O.; Vila, M. pg4DB: A fuzzy object-relational system. *Fuzzy Sets Syst.* **2008**, *159*, 1500–1514. [CrossRef]
38. Marin, N.; Medina, J.; Pons, O.; Sanchez, D.; Vila, M. Complex object comparison in a fuzzy context. *Inf. Softw. Technol.* **2003**, *45*, 431–444. [CrossRef]
39. Dubitzky, W.; Lopes, P.; White, J.; Anderson, J.M.C.C.; Dempsey, G.; Bell, D.A. A Holistic Approach to Coronary Heart Disease Risk Assessment Using Case-Based Reasoning. In Proceedings of the 2nd International Conference on Neural Networks and Expert Systems in Medicine and Healthcare, Plymouth, UK, 1 August 1996; pp. 97–103.
40. Bichindaritz, I.; Marling, C. Case-Based Reasoning in the Health Sciences: Foundations and Research Directions. In *Computational Intelligence in Healthcare 4*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 127–157. [CrossRef]
41. Montani, S.; Marling, C.; Bichindaritz, I.; Funk, P.; Lopez, B. CBR in the Health Sciences. In Proceedings of the CBR in the Health Sciences Workshop at the Twentieth International Conference on Case-Based Reasoning, Lyon, France, 3–6 September 2012.
42. Montani, S.; Striani, M. Artificial Intelligence in Clinical Decision Support: A Focused Literature Survey. *Yearb. Med. Inform.* **2019**, *28*, 120–127. [CrossRef]
43. Dubitzky, W.; Schuster, A.; Hughes, J.; Bell, D. An Advanced Case-Knowledge Architecture Based on Fuzzy Objects. *Appl. Intell.* **1997**, *7*, 187–204. [CrossRef]
44. Schuster, A.; Dubitzky, W.; Adamson, K.; Bell, D.A.; Hughes, J.G. Processing Similarity between a Mix of Crisply and fuzzily Defined Case Properties. In Proceedings of the 2nd International ICSC Symposium on Fuzzy Logic and Applications, Zurich, Switzerland , 17–19 September 1997; pp. 247–253.
45. Cai, B.; Zhang, Y.; Wang, H.; Liu, Y.; Ji, R.; Gao, C.; Kong, X.; Liu, J. Resilience evaluation methodology of engineering systems with dynamic-Bayesian-network-based degradation and maintenance. *Reliab. Eng. Syst. Saf.* **2021**, *209*, 107464. [CrossRef]
46. Hu, J.; Chen, C.; Liu, Z. Early warning method for overseas natural gas pipeline accidents based on FDOOBN under severe environmental conditions. *Process Saf. Environ. Prot.* **2022**, *157*, 175–192. [CrossRef]
47. Boulanger, D.; Colloc, J. Detecting Heterogeneity in a Multidatabase Environment through an O.O Model. In Proceedings of the IFIP, DS5, International Conference on Semantics of Interoperable Database Systems, Victoria, Australia, 16–20 November 1992.
48. Zighed D.A.; Duru, G.; Auray, J.-P. SIPINA, Méthode et Logiciel. 1992. Available online: https://www.librairie-levrailieu.fr/livre/9782905972279-sipina-methode-et-logiciel-zighed-jp/ (accessed on 10 December 2021).
49. Colloc, J.; Sybord, C. Représentation et Apprentissage des Processus de Décision en Médecine: Une Approche à Base de Cas. pp. 43–48. Available online: https://www.academia.edu/download/64665642/AIDRI97vsoumisefin.pdf (accessed on 10 December 2021).
50. Shen, Y.; Colloc, J.; Jacquet-Andrieu, A.; Lei, K. Emerging Medical Informatics with Case-Based Reasoning for Aiding Clinical Decision in Multi-Agent System. *J. Biomed. Inform.* **2015**, *56*, 307–317. [CrossRef]
51. Colloc, J.; Léry, N. A multi-expert decision support system in medical ethics. *Health Syst. Sci. Hermès* **1997**, *1*, 39–55.
52. Yameogo, R.A. Risques et Perspectives du Big Data et de L'intelligence Artificielle: Approche Ethique et Epistémologique. Ph.D. Thesis, University of Le Havre Normandy University, Le Havre, France, 2020.
53. Yameogo, R.A.; Colloc, J. Système multi-agents d'aide à la décision clinique, appliqué à la prise en charge de l'embolie pulmonaire. In Proceedings of the Applications Médicales de l'Informatique: Nouvelles Approches (AMINA'2016), Monastir, Tunisie, 30 November 2016; pp. 92–98, ISBN 978-9938-9531-2-1.
54. ESC. Guidelines for the diagnosis and management of acute pulmonary embolism in collaboration with the European Respiratory Society (ERS) The Task Force for the diagnosis and management of acute pulmonary embolism of the European Society of Cardiology (ESC). *Eur. Heart J.* **2020**, *41*, 543–603. [CrossRef]
55. Colloc, J.; Summons, P. An Analogical Model to Design Time in Clinical Objects. In Proceedings of the RITS, Dourdan, France, 25–27 March 2015; pp. 387–392.
56. Loubet, L. Les Maires Confrontés à L'Apprentissage de L'Intercommunalité: L'Exemple de L'Agglomération Toulousaine. Ph.D. Thesis, University Toulouse, Toulouse, France, 2011.
57. Loubet, L. L'apprentissage de la coopération intercommunale: Modalités et instruments. *L'Espace Polit.* **2012**, *18*. [CrossRef]
58. Ortony, A.; Clore, G.; Collins, A. *The Cognitive Structure of Emotions*; Cambridge University Press: Cambridge, UK, 1988.
59. Loubet, L. Emotions d'élus et découpage territorial. Recompositions intercommunales dans l'aire urbaine havraise. *L'Espace Polit.* **2020**. [CrossRef]
60. Colloc, J.; Summons, P. An Object Oriented Time Model of a Decision Support System for Intoxication Diagnoses. In Proceedings of the Industrial Simulation Conference, Palermo, Italy, 5–7 June 2006; pp. 387–392.
61. Shen, Y.; Colloc, J.; Jacquet-Andrieu, A.; Guo, Z.; Liu, Y. Constructing Ontology-based Cancer Treatment Decision Support System with Case-Based Reasoning. In Proceedings of the 2nd International Conference on Smart Computing and Communication SmartCom2017, Shenzhen, China, 10–12 December 2017.

62. Colloc, J.; Summons, P. An approach of the Process of Addiction: A model of the experience. In Proceedings of the Symposium on Reversible Time, Retardation and Anticipation in Quantum Physics, Biology and Cybernetics 29th Int Conference on Systems Research, Informatics and Cybernetics, Baden-Baden, Germany, 31 July–4 August 2017; Volume 1; pp. 103–107.
63. Colloc, J.; Yameogo, R.A.; Summons, P.F.; Shen, Y.; Park, M.; Aronson, J.E. EPICE an emotion fuzzy vectorial space for time modeling in medical decision. In Proceedings of the 1st International Conference on Internet of Things and Machine Learning, Liverpool, UK, 17–18 October 2017. [CrossRef]
64. Boulanger, D.; Colloc, J.; Chahwan, P. Dynamic Implementation of Composite Objects in ACSE Project. In Proceedings of the OOIS'94 International Conference on Object Oriented Information Systems, South Bank, London, UK, 19–21 December 1994; pp. 458–468.
65. Dori, D.; Tatcher, E. Selective multiple inheritance. *Softw. IEEE* **1994**, *11*, 77–85. [CrossRef]
66. Davies, P.C.W. The epigenome and top-down causation. *Interface Focus* **2011**, *2*, 42–48. [CrossRef] [PubMed]
67. Ensing, M.; Paton, R.; Speel, P.H.; Rada, R. An object-oriented approach to knowledge representation in a biomedical domain. *Artif. Intell. Med.* **1994**, *6*, 459–482. [CrossRef]
68. Cox, B. *Object-Oriented Programming: An Evolutionary Approach*; Addison-Wesley: Reading, MA, USA, 1986.
69. Meyer, B. *Object-Oriented Software Construction*; Prentice-Hall: New York, NY, USA, 1988.
70. Gamma, E.; Helm, R.; Johnson, R.; Johnson, R.E.; Vlissides, J. *Design Patterns—Elements of Reusable Object-Oriented Software*; Addison-Wesley Publishing Company: Boston, MA, USA, 1995.
71. Shalloway, A.; Trott, J.R. *Design Patterns Explained: A New Perspective On Object-Oriented Design*; Addisson Wesley: Boston, MA, USA, 2002.
72. Boulanger, D.; Colloc, J.; Talens, G.; Oussalah, C. An Object Oriented Reuse Repository. In *KBUP95 First International Workshop on, Knowledge-Based Systems for the (Re) Use of Program Libraries*; Opengrey: Nancy, France, 1995; pp. 86–94.
73. Delgado, M.; Marín, N.; Pérez, Y.; Vila, M.A. Bipolar queries on fuzzy univalued and multivalued attributes in object databases. *Fuzzy Sets Syst.* **2016**, *292*, 175–192. [CrossRef]
74. Zhang, F.; Ma, Z.; Yan, L.; Wang, Y. A description logic approach for representing and reasoning on fuzzy object-oriented database models. *Fuzzy Sets Syst.* **2012**, *186*, 1–25. [CrossRef]
75. Cao, T.; Rossiter, J. A deductive probabilistic and fuzzy object-oriented database language. *Fuzzy Sets Syst.* **2003**, *140*, 129–150. [CrossRef]
76. Ebrahiminejad, M.; Shakeri, E.; Ardeshir, A.; Zarandi, M.F. An object-oriented model for construction method selection in buildings using fuzzy information. *Energy Build.* **2018**, *178*, 228–241. [CrossRef]
77. Jain, S.; Seeja, K.; Jindal, R. A fuzzy ontology framework in information retrieval using semantic query expansion. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 100009. [CrossRef]
78. Calegari, S.; Ciucci, D. *Fuzzy Ontology, Fuzzy Description Logics and Fuzzy-OWL*; Springer: Berlin/Heidelberg, Germany, 2007. [CrossRef]
79. Pedrycz, W.; Sosnowski, Z. Fuzzy object-oriented system design. *Fuzzy Sets Syst.* **1998**, *99*, 121–134. [CrossRef]
80. Cross, V.V. Fuzzy extensions for relationships in a generalized object model. *Int. J. Intell. Syst.* **2001**, *16*, 843–861. [CrossRef]