Cognitive Map Query Language for Temporal Domains

Adrian Robert, David Genest(✉), and Stéphane Loiseau

LERIA, Université d'Angers, Angers, France
{adrian.robert,david.genest,stephane.loiseau}@univ-angers.fr

Abstract. This article introduces the temporal cognitive maps model and its associated query language.

A cognitive map is a graph used to model strategies or influence systems. Each node represents a concept and each edge represents an influence.

One limit of cognitive maps is that temporal features cannot be taken account in the model.

This article proposes an extended model of cognitive map, called temporal cognitive maps, that includes temporal features in a cognitive map.

This article also proposes a temporal cognitive map query language that accesses all the components of a temporal cognitive map: concept, influence and temporal features.

Keywords: Time representation · Cognitive map · Query language · OWL

1 Introduction

This article is an improved and extended version of [27]. These extensions have been made: *(i)* bibliography in cognitive maps, temporal models and query languages are extended, *(ii)* seven primitives, instead of two, are presented in detail, *(iii)* the TCMQL language, along with its syntax and semantics is presented.

The *cognitive map* [1] model is a semantic model coming from cognitive psychology. This model represents strategies expressed in an influence system. A cognitive map is an oriented graph whose nodes are labeled by *concepts* and edges, called *influences*, are labeled by an *influence value*. Influence values belong to a predefined value set. A sequence of influences from a node to another makes a *path*. The model can infer a *propagated influence value* from a node to another. A *taxonomic cognitive map* [18] is a cognitive map defined on a *taxonomy*. The taxonomy organizes the concepts with'kind of' type relations: the nodes of the taxonomic cognitive map are labeled by concepts of the taxonomy.

The Kifanlo project aimed to study the evolution of the fishing strategies in the Atlantic coast from 1970 to 2016. About fifty cognitive maps have been designed with fishermen to model their fishing strategies. Half of those maps represents fishing strategies in the seventies, the other half represent current

A. P. Rocha et al. (Eds.): ICAART 2020, LNAI 12613, pp. 215–235, 2021.
https://doi.org/10.1007/978-3-030-71158-0_10

fishing strategies, each map contains 25 to 50 nodes. A cognitive map edition software, VSPCC, has been used and improved[1]. In the Kifanlo project, there is a significant number of concepts that have a temporal semantics. These concepts usually repeat periodically over time like seasons, fishing seasons and so on... This periodicity of the concepts should be taken into account in cognitive maps; that was not the case in VSPCC. So, this paper introduces temporal cognitive maps, which is a new model that extends taxonomic cognitive maps with a temporal ontology for representation and reasoning. No previous works including temporal aspects in concepts of cognitive maps has been proposed in bibliography.

As seen in [27], because of the periodicity of the concept's semantics, the *temporal ontology* aims to represent *periodic intervals* [22]. It uses *temporal assertions* that are triples made of two periodic intervals related by a *comparison predicate*. A *temporal cognitive map* is defined on a temporal ontology; it contains a set of temporal assertions that link the nodes of the cognitive map to the temporal ontology. The nodes can thus be temporally characterized, meaning that a certain influence holds with respect to the temporal assertions of its nodes. To reason with a temporal cognitive map, this article presents the 'Temporal Cognitive Map Query Language' *TCMQL*, which is an extension of the query language for cognitive maps CMQL [25]. TCMQL is made with seven primitives which two of them are temporal *primitives* presented in [27]: *TimeInfo* and *Compare*. *TimeInfo* lets the user access the periodic interval associated with a node. *Compare* infers new information using temporal assertions of nodes and the temporal ontology. This extension provides a way to use the temporal information of the model for a further analysis of cognitive maps. TCMQL, as well as VSPCC extended to temporal cognitive maps, have been delivered to the researchers in geography that work in the Kifanlo project for further analysis[2].

The article is composed of five parts. The first is the bibliographic part. The second recalls the taxonomic cognitive map model. The third introduces the temporal cognitive map model. The fourth presents primitives. The fifth describes the TCMQL language.

2 Bibliography

The bibliography related with cognitive map query language for temporal domains can be connected to three kinds of works. First and principally, the works that focus on cognitive maps. Second, the numerous works that are related on temporal models. Third, the research works about query languages.

Cognitive maps and Bayesian networks [23] are two models that stem on causal graphical networks. While Bayesian networks focus on the computation of influences based on conditional probabilities, cognitive maps focus mainly on the visualization. So cognitive maps are easier to understand for several types

[1] VSPCC [19] has been implemented after the thesis of Aymeric LeDorze [17], for the project Kifanlo.

[2] This work is being led in the project *Analyse Cognitive de Savoirs (ACS)* granted by the french region Pays de la Loire from 2017 to 2020.

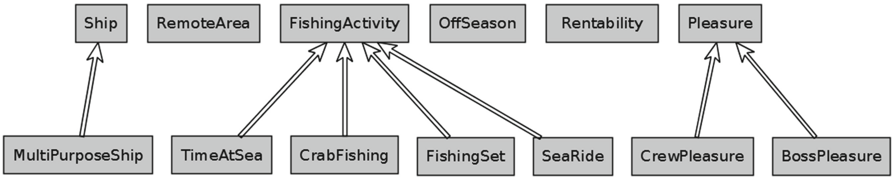

Fig. 1. A taxonomy $\mathcal{T}_1$ [27].

of user. The semantic of the influence, called propagated influence in cognitive map model can vary depending on the subject and users. It also can be seen as a drawback of cognitive maps versus Bayesian networks that stem always on the same theoretical basis. In cognitive map models, depending on the problem, various sets of influence values can be used: $\{-,+\}$ and $[-1,1]$ are the most used; there are also $\{none, some, much, alot\}$ [15] or $\{-4,-3,-2,-1,0,1,2,3,4\}$ [17]. Cognitive maps are used in many fields such as social sciences [1], biology [21] and geography [4].

An important domain for artificial intelligence research is a study of temporal concepts. Many aspects of time exist, and many solutions to model them and make some inferences with them have been proposed. No cognitive map model takes into account temporal concepts, but few articles integrates temporal aspects on the influences: they explain the delay needed to make an influence from a concept to another one, for instance [3,29] provides a way to describe an approximation of the time needed for making an influence. For our applications, the core of the temporal aspects needed to be modelled are periodic. A periodic interval [7,8,22] is a type of non-convex interval [16], which is an interval composed of several unconnected convex subintervals. Periodic intervals have the particularity to be composed of subintervals that have the same length and are equally spaced. For instance 'winter' is a periodic interval.

The relational model [6] is the standard model to manage databases. Relational databases are queried with the popular query language SQL that is a standard; SQL [14] is known by many people, including non-engineer's ones. So, the syntax of SQL provides the bases of many query languages for databases or knowledge bases. It is the case of SPARQL [11] that is the language used to query RDF knowledge bases; it is also the case for query languages for property graphs, like GraphQL, PGQL, Cypher [9]. There is no query language to query cognitive maps, except CMQL [26] that is the base of TCMQL, the query language presented in this paper. CMQL is a query language whose syntax is close to the one of SQL and whose semantics is similar to the one of the domain relational calculus [20,26]. CMQL's particularity resides in the use of many primitives that allow to access the various features of a taxonomic cognitive map set.

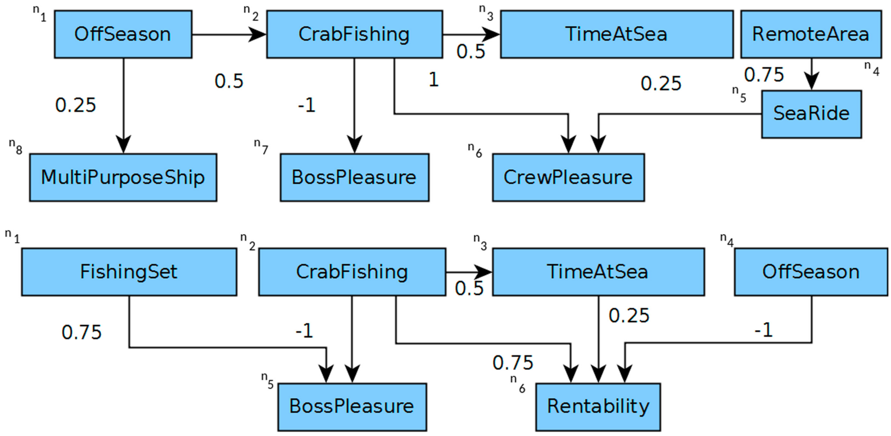

Fig. 2. Two taxonomic cognitive maps, $CC1$ (top) and $CC2$ (bottom) [27].

3 Taxonomic Cognitive Map

A taxonomic cognitive map is a graph whose nodes and edges are respectively labeled by a concept of a taxonomy and by an influence value; the taxonomy aims to organize the concepts. The taxonomy is even more useful when using a set of cognitive maps, to make sure that different cognitive maps use the same concepts [5].

3.1 Taxonomic Cognitive Map Model

The taxonomy organizes the concepts by specifying a specialization relation between them

Definition 1 (Taxonomy). *Let C be a concept set. A* taxonomy $\mathcal{T} = (C, \leq)$ *is a set of rooted trees of concepts that represents a partial order relation $\leq$ whose meaning is 'kind of'.*

Example 1. $\mathcal{T}_1$ is the taxonomy of the Fig. 1. Some concepts are ordered by a relation of specialization. For instance, the relation *MultiPurposeShip* $\leq$ *Ship*, meaning that *MultiPurposeShip* is a kind of *Ship*, is represented by an arrow in the figure.

The most specialized concepts of the taxonomy are said elementary.

Definition 2 (Elementary Concepts). *Let $\mathcal{T} = (C, \leq)$ be a taxonomy. The* elementary concepts *of $\mathcal{T}$ are: $elem_{\mathcal{T}} = \{c \in C / \ \forall c' \in C, c' \leq c \implies c' = c\}$.*

Example 2. In $\mathcal{T}_1$, the elementary concepts are $elem_{\mathcal{T}_1} = \{$*MultiPurposeShip*, *RemoteArea*...$\}$; only the concepts *Ship*, *FishingActivity* and *Pleasure* are not elementary.

A taxonomic cognitive map is a graph whose nodes and edges are respectively labeled by an elementary concept of a taxonomy and an influence value. The influence value represents the strength of the influence and belongs to a defined value set which can be qualitative or quantitative, discrete or continuous.

Definition 3 (Taxonomic Cognitive Map). *A* taxonomic cognitive map *defined on a taxonomy* $\mathcal{T} = (C, \leq)$ *and a value set* I *is an oriented labeled graph CM = (N,E,labelN, labelE) such that:*

- *N : the nodes of the graph.*
- $E \subseteq N \times N$*: the edges are called influences.*
- $labelN : N \rightarrow elem_{\mathcal{T}}$ *is a label function on the nodes.*
- $labelE : E \rightarrow I$ *is a label function on the edges.*

Example 3. $CC1$ and $CC2$ are the two taxonomic cognitive maps of the Fig. 2. They are defined on the taxonomy $\mathcal{T}_1$ of the Fig. 1 and the value set $I = [-1, 1]$. Note that, in the figure, each node has a unique identifier per map n_1, n_2... that is displayed only for clarity in this paper. An influence labeled by 1 (resp. -0.25) means that the source node influences strongly (resp. weakly) and positively (resp. negatively) the destination node. In our application, each fisherman designs a cognitive map: $CC1$ has been designed by *fisherman1*, and $CC2$ by *fisherman2*. In $CC2$, the node n_2 (*CrabFishing*) influences strongly and negatively (-1) the node n_5 (*BossPleasure*); which means that the boss does not like fishing crab.

3.2 Taxonomic Cognitive Map Inference

A path is a sequence of influence which represents a way a node of the map influences another. A path is said minimal if it does not contain any cycle. Notice that between two nodes, there can be more than one minimal path.

Definition 4 (Path, *source*, *dest*, $Paths_{CM}$). *Let CM =(N,E,labelN,labelE) be a taxonomic cognitive map defined on* $\mathcal{T} = (C, \leq)$ *and I. Let a,b* $\in N$ *be two nodes of CM.*

- *A* path *P from a to b is a sequence of length* $length_P \geq 1$ *of influences* $(u_i, u_{i+1}) \in E$ *(with* $i \in [0; length_P - 1]$*) such that* $a = u_0$ *is the* source *of P and* $b = u_{length_P}$ *is the* destination *of P. This path is denoted by* $a \rightarrow u_1 \rightarrow \cdots \rightarrow b$.
- $source_P = labelN(a)$ *is the* source concept *of P and* $dest_P = labelN(b)$ *is the* destination concept *of P.*
- *A path P is said* minimal *if* $\forall i, j \in [0; length_P], i \neq j \Rightarrow u_i \neq u_j$.
- *The* set of all minimal paths *on CM is denoted by* $Paths_{CM}$. *Let S be a set of taxonomic cognitive maps, the* set of paths on *S, denoted by* $Paths_S$*, is defined as* $Paths_S = \bigcup_{s \in S} Paths_s$.

Example 4. This example is based on $CC2$ (Fig. 2). $p_1 = n_2(CrabFishing) \rightarrow n_3$ $(TimeAtSea) \rightarrow n_6$ $(Rentability)$ is a minimal path from the source node n_2 to the destination node n_6, $length_{p_1} = 2$. $p_2 = n_2$ $(CrabFishing) \rightarrow n_6$ $(Rentability)$ is a minimal path, $length_{p_2} = 1, source_{p_2} = CrabFishing$.

One of the main features of cognitive maps is their ability to infer the propagated influence from any node to any other one, which denotes a value of influence. To do that, every influence path from the node to the other is involved. The propagated influence from a node to another can be calculated differently depending on the map's semantics and on the value set on which it is defined. In all cases, the computation of the propagated influence first assigns a path value for each path with a function, then secondly aggregates those values with an other function.

Definition 5 (Propagated Influence, PV). *Let CM=(N,E,labelN,labelE) be a taxonomic cognitive map defined on $\mathcal{T} = (C, \leq)$ and I.*

- *The* path value *is a function $PV_{path}: Paths_{CM} \rightarrow I$ which infers the propagated influence of a path.*
- *The* propagated influence value *is a function $PV: N \times N \rightarrow I$ which infers the propagated influence from a node to another one, aggregating the path values of each path between the two nodes.*

In this paper, we will use the value set $I = [-1, 1]$. A product function will be used as path value and a mean function for the propagated influence value as it is often done in cognitive maps [10].

Example 5. This example is based on $CC2$ (Fig. 2). The paths p_1 and p_2 come from the example 4. Let's infer the propagated influence value between n_2 and n_6, respectively labeled by *CrabFishing* and *Rentability*. The set of all minimal paths between those two nodes is $\{p_1, p_2\}$. To infer the propagated influence value between n_2 and n_6 we need $PV_{path}(p_1)$ and $PV_{path}(p_2)$. From the chosen product function, we have $PV_{path}(p_1) = 0.5 * 0.25 = 0.125$ and $PV_{path}(p_2) = 0.75$. Then, aggregating the path values, $PV(n_2, n_6) = \frac{(0.125 + 0.75)}{2} = 0.44$. So the propagated influence value from n_2 to n_6 is 0.44.

The taxonomic cognitive map model can also infer a taxonomic influence value which is used to infer the influence value between any pair of concepts of the taxonomy. Note that the propagated influence value is a particular case of the taxonomic influence value where the concepts are elementary. The taxonomic influence value is not presented in this article, but is described in [5].

4 Time Representation

This section introduces the periodic intervals, then proposes a temporal ontology defined on those periodic intervals and temporal assertions that compare pairs of them. So, the temporal cognitive map can be introduced, it is a taxonomic cognitive map defined on a temporal ontology.

4.1 Periodic Intervals

The periodic intervals of Osmani and Balbiani [2,22] that also considers qualitative relations between them are chosen. This approach is relevant to the Kifanlo project and, in general, seems suited for cognitive maps as it offers more flexibility and handles the lack of precise information.

Definition 6 (Periodic Interval). *A* periodic interval *is a non-convex interval whose subintervals are equally spaced and have equal length.*

Example 6. January is a periodic interval since all its subintervals last one month and occur every year. *Summer* is also a periodic interval, with subintervals lasting three months and occurring every year.

This paper proposes to specify those periodic intervals with qualitative relations between two intervals using a comparison predicate. Those predicates are the 16 relations of Osmani [22] plus 5 relations. The relations of Osmani are very similar to the 13 relations of the Allen's intervals, except that the precedence and its inverse are replaced by 5 relations which consider the periodicity. This paper also considers two relations (*inside*/*disjoint*) that combine some of Osmani's relations and three relations (<,>,=) that compare duration of intervals, which can not be done with Osmani's intervals.

Definition 7 (Comparison Predicate). *A* comparison predicate *is a binary relation whose domain and range are periodic intervals.* $\mathcal{P}$ *is the set of the 21 comparison predicates: {m,mi,s,si,d,di,f,fi,o,oi,eq,ppi,mmi,moi,omi,ooi,in,dis,<, =,>}*

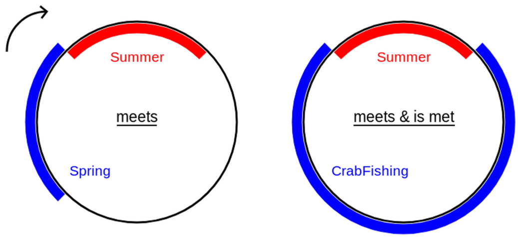

Fig. 3. Two cyclic representations of relations between periodic intervals [27].

The table below shows the 16 relations of Osmani & Balbiani, the column *meaning* explains the relations through an ordering of the boundaries ($A1$,$A2$,$B1$,$B2$) of the periodic intervals A and B. This ordering comes from the CYCORD theory [28]. Two added relations are: '*in*' (Inside) which is the disjunction of '*s*', '*d*', '*f*', '*eq*' and '*dis*' (Disjoint) which is the disjunction of '*m*', '*mi*', '*mmi*', '*ppi*'.

To these relations are also added three relations to compare the duration of periodic intervals: '<', '>' and '='.

Periodic intervals and comparison predicates defined above are used to represent temporal knowledge through temporal assertions. A temporal assertion is an assertion which represents a relation between two periodic intervals. It is a triple (*interval, predicate, interval*).

Definition 8 (Temporal Assertion). *$\mathcal{P}$ is the set of the 21 comparison predicates. A* temporal assertion *is an assertion which constitute a triple* (e_1, p, e_2) *such that* $p \in \mathcal{P}$ *and* e_1 *and* e_2 *are periodic intervals.*

Example 7. Relations between periodic intervals are often represented on a circle (Fig. 3) which is to be read clockwise. The first circle represents the temporal assertion (*Spring, meets, Summer*) and it matches the ordering ('*SpringBegins*', '*SpringEnds*' = '*SummerStarts*', '*SummerEnds*') of the second row of the table. Its inverse relation is *mi* (*is met by*), so we have (*Summer, is met by, Spring*). The second circle illustrates the temporal assertion (*CrabSeason, meets & is met, Summer*). *CrabSeason* is related to *Summer* by the relation '*meets & is met*' which means that the crab season starts when summer ends and ends when summer starts. Some comparison predicates are used to compare duration, for instance in the temporal assertion (*Day, <, Month*) the comparison predicate '<' is used to compare the duration of *Day* and *Month*.

name	*meaning*	*inverse*
eq (*equals*)	$A1 = B1\ ,\ A2 = B2$	(*eq*)
m (*meets*)	$A1\ ,\ A2 = B1\ ,\ B2$	**mi**
s (*starts*)	$A1 = B1\ ,\ A2\ ,\ B2$	**si**
d (*during*)	$A1\ ,\ A2\ ,\ B2\ ,\ B1$	**di**
f (*finishes*)	$A1\ ,\ A2 = B2\ ,\ B1$	**fi**
o (*overlaps*)	$A1\ ,\ B1\ ,\ A2\ ,\ B2$	**oi**
ppi (*precedes & is preceded*)	$A1\ ,\ A2\ ,\ B1\ ,\ B2$	(*ppi*)
mmi (*meets & is met*)	$A1 = B2\ ,\ A2 = B1$	(*mmi*)
moi (*meets & is overlapped*)	$A1\ ,\ B2\ ,\ A2 = B1$	**omi**
ooi (*overlaps & is overlapped*)	$A1\ ,\ B2\ ,\ B1\ ,\ A2$	(*ooi*)
in (*inside*)	$s \vee d \vee f \vee eq$	
dis (*disjoint*)	$m \vee mi \vee mmi \vee ppi$	
< (*is shorter*)	$\overline{A1A2} < \overline{B1B2}$	>
= (*has same length*)	$\overline{A1A2} = \overline{B1B2}$	(=)

Fig. 4. Meaning of comparison predicates [27].

4.2 Time Ontology

Many temporal ontologies exist, amongst those, OWL-Time ontology [13] is a W3C reference and one of the most used. It turns out that time ontologies do not take into account periodic intervals and certainly not the qualitative relations to compare them. That is why this paper introduces a new temporal ontology that considers periodic intervals and could be added to existing heavier temporal ontologies like OWL-Time. Our light-weight temporal ontology is composed of the class *PeriodicInterval*, the 21 comparison predicates as object properties, a set of instances of *PeriodicInterval* and a set of temporal assertions on these individuals.

Definition 9 (Temporal Ontology). *A temporal ontology* $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$ *is an ontology such that:*

- $\mathcal{P}$ *is the set of the comparison predicates.*
- $\mathcal{E}$ *is a set of periodic intervals.*
- $\mathcal{A}$ *is a set of temporal assertions of the ontology.*

Example 8. The Fig. 5 represents the temporal ontology $\mathcal{O}_1$. The periodic intervals of this ontology are $\mathcal{E}$=$\{$*Spring, CrabSeason, Year* $\ldots\}$ and the temporal assertions are $\mathcal{A} = \{$(*Season*,<,*Year*), (*CrabSeason*, *meets & is met*, *Summer*) $\ldots\}$.

4.3 Temporal Cognitive Map

A temporal cognitive map is a taxonomic cognitive map defined on a temporal ontology. Each node of the map is labeled by a periodic interval and a set of temporal assertions links those periodic intervals to the ontology. This way, nodes may be temporally characterized.

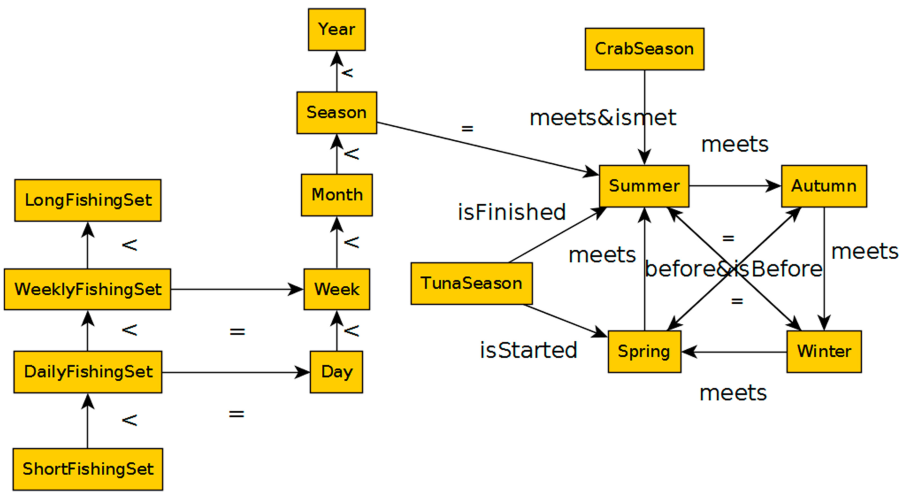

Fig. 5. A partial representation of the temporal ontology $\mathcal{O}_1$ [27].

Definition 10 (Temporal Cognitive Map). *Let* $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$ *be a temporal ontology, let* $T = (C, \leq)$ *be a taxonomy and* I *a value set. A* temporal cognitive map *TCM defined on* $\mathcal{O}$ *is a sextuplet* $(N, E, labelN, labelE, labelT, \mathcal{A}_{TCM})$ *such that:*

- $(N, E, labelN, labelE)$ *is a taxonomic cognitive map defined on* $\mathcal{T}(C, \leq)$ *and* I*.*
- $labelT$ *is a label function on the nodes of the map which attaches a unique periodic interval* e_n *to a node* n*.*
- $\mathcal{A}_{TCM}$ *is a set of temporal assertions* (e_1, p, e_2) *where* $labelT^{-1}(e_1) \in N$ *and* $e_2 \in \mathcal{E}$*.*

Example 9. This example[3] describes the two temporal cognitive maps of the Fig. 6: $TCM1$ and $TCM2$. A temporal assertion (in yellow) of a temporal cognitive map is visually represented below the node (in blue) that it characterizes. The periodic interval attached to the node is visually omitted, that is why temporal assertions are written as couples and not triples. For instance in $TCM1$, the node labeled by *OffSeason* is characterized by the temporal assertion (*TCM1_ OffSeason, si, Summer*) where *TCM1_ OffSeason* is the omitted periodic interval attached to this node and '*si*' is the comparison predicate '*isStartedBy*'. Notice that several temporal assertions can be attached to the same node,

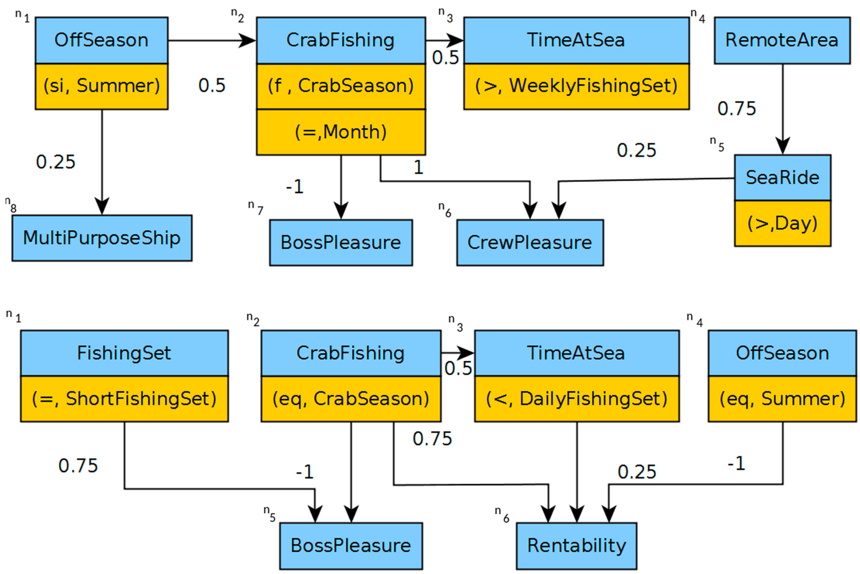

Fig. 6. Two temporal cognitive maps, $TCM1$ (top) and $TCM2$ (bottom) [27].

[3] A set of temporal cognitive maps based on the same taxonomy, value set and temporal ontology will be considered. Since many maps may contain nodes labeled by the same concept, the following notation is used: The periodic interval associated with a node labeled by a concept 'c' of a map 'm' is noted 'm_c'.

as it is the case for the node labeled by *CrabFishing* in $TCM1$. This node is characterized by a periodic interval that lasts one month (=, *Month*) at the end of the *CrabSeason* (*f, CrabSeason*). *fisherman1* fishes crab for one month at the end of the crab season.

5 Primitives

In this section, we first define what a primitive is and how it can be used. Then we recall the basic primitives of the model, concerning the presence of concepts in a map (*IsInMap*), taxonomy (*KindOf*), paths (*Path, PathValue*) and influences (*InfluenceValue*). Finally, we present in detail primitives concerning the temporal aspects: *TimeInfo* and *Compare*, which allow to access the concepts' temporal assertions and compare them.

5.1 Primitive and Primitive Formula

A primitive is defined as a relation indexed by its attributes, definitions and notations are based on those of the domain relational calculus: an element of a relation, i.e. a tuple, is seen as a mapping from the attributed of the relation to values of the domains.

Definition 11 (Primitive). *Let* $A = \{A_1, \dots A_n\}$ *be a set of attributes and* $D = \{D_1, \dots D_n\}$ *be a set of domains. A* primitive *is a relation* $R(A_1 : D_1, \dots, A_n : D_n)$ *of arity* n*, its value is a subset of the cartesian product of its domains indexed by its attributes :* $val(R) \subseteq \{t : A \rightarrow D_1 \cup \dots \cup D_n / \forall i \in [1, n] t(A_i) \in D_i\}$.

In order to provide examples in this section, we also introduce primitive formulas. A primitive formula can be seen as a very simple query and will be used in the condition clause of TCMQL queries. A primitive formula is a syntactic construction based on a primitive, whose arguments are the attributes of the primitive, they can be variables or constants.

Definition 12 (Primitive Formula). *A* primitive formula f *on a* n*-ary primitive* P *is an expression* $P(X_1, \dots X_n)$ *where each* X_i*, called term, is either a variable, i.e. a syntactic expression prefaced with "?" or a constant.*

Let $f = P(X_1, \dots, X_n)$ *be a primitive formula, let* m *be the number of variables in* $X_1 \dots X_n$*. The* value *of* f *is the set of tuples of arity* m *:* $\{t / \exists t_1 \in val(P), \forall i \in [1, n]\ t_1(A_i) = X_i \textit{ if } X_i \textit{ is a constant; } t_1(A_i) = t(X_i) \textit{ if } X_i \textit{ is a variable}\}$.

5.2 The Path Primitive

The *Path* primitive links a path with its source concept and destination concept, it provides an access to paths and concepts.

Definition 13 (Path). *Let S be a set of temporal cognitive maps defined on the same temporal ontology $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$, taxonomy $T = (C, \leq)$ and I. Path*($m : S, c_1 : C, c_2 : Cp : Paths_S$) *is a primitive whose value is* $\{(m, c_1, c_2, p) / \ m \in S, p \in Paths_m, source_p = c_1$ *and* $dest_p = c_2\}$.

Example 10. – The following examples uses the maps $TCM1$ and $TCM2$ (Fig. 6).

– *Path(TCM2,CrabFishing,Rentability,?p)* is a primitive formula. Its value is a unary relation whose set of tuples (?p) is:

$?p$
$CrabFishing \rightarrow TimeAtSea \rightarrow Rentability$
$CrabFishing \rightarrow Rentability$

– *Path(TCM2,CrabFishing,?c2,?p)* is a primitive formula. It uses $TCM2$. Its value is a relation whose set of tuples (?c2,?p) is:

$?c2$	$?p$
$TimeAtSea$	$CrabFishing \rightarrow TimeAtSea$
$Rentability$	$CrabFishing \rightarrow TimeAtSea \rightarrow Rentability$
$Rentability$	$CrabFishing \rightarrow Rentability$
$BossPleasure$	$CrabFishing \rightarrow BossPleasure$

5.3 The PathValue Primitive

The primitive *PathValue* links a path with its value.

Definition 14 (PathValue). *Let S be a set of temporal cognitive maps defined on the same temporal ontology $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$, taxonomy $T = (C, \leq)$ and I. PathValue*($m : S, p : Paths_S, i : I$) *is a relation whose value is* $\{(m, p, i)/m \in S, p \in Paths_m$ *and* $i = PV_{path}(p)\}$.

Example 11. PathValue*(TCM2,CrabFishing → TimeAtSea → Rentability, ?i)* is a primitive formula. Its value is a unary relation whose set of tuples ($?i$) is:

$?i$
0.125

5.4 The InfluenceValue Primitive

The primitive *InfluenceValue* links two concepts of the taxonomy with the propagated influence value from the first to the second.

Definition 15 (InfluenceValue). *Let S be a set of temporal cognitive maps defined on the same temporal ontology $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$, taxonomy $T = (C, \leq)$ and I. InfluenceValue*($m : S, c_1 : C, c_2 : C, i : I$) *is a relation whose value is* $\{(m, c_1, c_2, i)/m = (N, E, labelN, labelE, labelT, \mathcal{A}_m) \in S, c_1, c_2 \in C, \exists n_1, n_2 \in N, labelN(n_1) = c_1, labelN(n_2) = c_2 \text{ and } i = PV(n_1, n_2)\}$.

Example 12. InfluenceValue(TCM2, CrabFishing, Rentability,?i) is a primitive formula. Its value is a unary relation whose set of tuples is:

$?i$
0.44

5.5 The IsInMap Primitive

The primitive *IsInMap* links a map and a concept of this map.

Definition 16 (IsInMap). *Let S be a set of temporal cognitive maps defined on the same temporal ontology $\mathcal{O}$, taxonomy $T = (C, \leq)$ and I. IsInMap*($m : S, c : C$) *is a relation whose value is* $\{(m, c)/m = (N, E, labelN, labelE, labelT, \mathcal{A}_m) \in S, \exists n \in N \text{ such that } labelN(n) = c\}$.

Example 13. IsInMap(?m, Rentability) is a primitive formula querying for maps that contains the concept *Rentability*. Its value is a unary relation whose set of tuples is:

$?m$
$TCM2$

5.6 The KindOf Primitive

KindOf is a binary primitive with two concepts of the taxonomy where the first is a kind of the second.

Definition 17 (KindOf). *Let $T = (C, \leq)$ be a taxonomy. KindOf*($c_1 : C, c_2 : C$) *is a relation whose value is* $\{(c1, c2)/c_1 \leq c_2\}$.

Example 14. KindOf(?c, Pleasure) is a primitive formula querying for specializations of *Pleasure*. Its value is a unary relation whose set of tuples is:

$?c$
$Pleasure$
$CrewPleasure$
$BossPleasure$

5.7 The TimeInfo Primitive

The extraction primitive *TimeInfo* links a cognitive map, a concept of this map, and the periodic interval associated with the node labeled by this concept in this map.

Definition 18 (TimeInfo). *Let S be a set of temporal cognitive maps defined on the same temporal ontology* $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$*, taxonomy* $T = (C, \leq)$ *and I. Let* $\mathcal{E}_S$ *be the set of all periodic intervals associated with the nodes of the maps in S. The primitive* TimeInfo$(m : S, c : C, i : \mathcal{E}_S)$ *is a relation whose value is* $\{(m, c, i)/m = (N, E, labelN, labelE, labelT, \mathcal{A}_m) \in S, \exists n \in N, labelT(n) = i$ *and* $labelN(n) = c\}$

Example 15. TimeInfo(?map,TimeAtSea,?interval) is a primitive formula. Its value is a binary relation whose value is the set of tuples *(?map,?interval)* in which *?interval* is associated with the node labeled by the concept *TimeAtSea* in *?map*:

?map	?interval
$TCM1$	$TCM1\_TimeAtSea$
$TCM2$	$TCM2\_TimeAtSea$

The primitive *TimeInfo* is used here to link concepts and maps to associated periodic intervals, *TimeAtSea* is then used in $TCM1$ and $TCM2$.

Used alone the usefulness of this primitive is limited, it is often used in conjunction with the primitive *Compare* defined below.

5.8 The Compare Primitive

When the designer of a temporal cognitive map adds his domain knowledge, he adds the least amount of temporal assertions and expects the implicit ones to be taken into account: an inference is thus necessary. 151 inference rules are used for these inferences, they are OWL2 [12] rules. The comprehensive list of rules is not given in the paper as it is too long but available online [24]. The rules about the 16 Balbiani's relations can be found also in the references [2], a few other rules about the new predicates are added.

Example 16. Here are some inference rules:

- *SubObjectPropertyOf(during inside)* which means (e_1, *during*, e_2)→(e_1, *inside*, e_2)
- *SubObjectPropertyOf(starts <)* which means (e_1, *starts*, e_2)→(e_1, <, e_2)
- *SubObjectPropertyOf(ObjectPropertyChain(meets startedBy) meets)* which means (e_1, *meets*, e_2) ∧ (e_2, *startedby*, e_3) → (e_1, *meets*, e_3)

Using the ontology of the Fig. 5 and the cognitive maps Fig. 6, new assertions are inferred:

- *(CrabSeason, disjoint, Summer)* which means that the crab season is outside the summer. This assertion comes from the assertion *(CrabSeason, mmi, Summer)* of $\mathcal{O}_1$ and the rule (e_1, mmi, e_2)→(e_1, disjoint, e_2).
- *(TunaSeason, >, Season)* which means that the season of tuna is longer than a calendar season. This assertion comes from the assertions *(TunaSeason, fi, Summer)*, *(Season, =, Summer)* of $\mathcal{O}_1$ and the rules (e_1, fi, e_2)→(e_1, <, e_2) and (e_1, >, e_2) ∧ (e_2,=, e_3) → (e_1, >, e_3).
- *(TCM1_ CrabFishing, meets, Summer)* which means that Summer starts when the crab season ends. This assertion comes from the assertions *(TCM1_ CrabFishing, f, CrabSeason)* of TCM1, *(CrabSeason, mmi, Summer)* of $\mathcal{O}_1$ and the rule (e_1, f, e_2) ∧ (e_2, mmi, e_3) → (e_1, meets, e_3).

Inferences can be carried out on a set that contains the temporal assertions of the ontology and the temporal assertions of each temporal cognitive map. The saturated set is the set of temporal assertions that can be deduced from all these temporal assertions and the inference rules.

Definition 19 (Saturated Set). *Let $\mathcal{R}$ a set of rules and $S = \{(CM_1, labelT_1, \mathcal{A}_1), \ldots, (CM_k, labelT_k, \mathcal{A}_k))\}$ be a set of k temporal cognitive maps defined on $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$, $\mathcal{I}_S$ is the* saturated set of temporal assertions *resulting from the inference of the rules of $\mathcal{R}$ on the set $\mathcal{A} \cup \bigcup_{i=1}^{k} \mathcal{A}_i$.*

The primitive *Compare* uses the saturated set of temporal assertions, it is a relation between two periodic intervals and a comparison predicate which is a valid comparison between these intervals.

Definition 20 (Compare). *Let S be a set of temporal cognitive maps defined on the same ontology $\mathcal{O} = (\mathcal{P}, \mathcal{E}, \mathcal{A})$ where $\mathcal{P}$ is the set of comparison predicates. Let $\mathcal{E}_S$ be the set all periodic intervals associated with the nodes of the maps in S. Let $\mathcal{I}_S$ the saturated set of all temporal assertions.*

The primitive Compare*(*e1: $\mathcal{E}_S \cup \mathcal{E}$,p: $\mathcal{P}$,e2: $\mathcal{E}_S \cup \mathcal{E}$*) is a relation whose value is $\{(e_1, p, e_2)/(e_1, p, e_2) \in \mathcal{I}_S\}$.*

Example 17. The following examples use the ontology $\mathcal{O}_1$ (Fig. 5) and the temporal cognitive maps $TCM1$ and $TCM2$ (Fig. 6).

- *Compare(TunaSeason, ?pred, ?interval)* is a primitive formula which aims to compare the periodic interval *TunaSeason* (which is from *Spring* to *Summer* according to $\mathcal{O}_1$) to any other periodic interval. There are many result tuples like *(>, Summer)* since the *Summer* finishes the *TunaSeason*:

$?pred$	$?interval$
$isStarted$	$Spring$
$isFinished$	$Summer$
$>$	$Summer$
$>$	$Week$
$isFinished$	$TCM2\_OffSeason$
...	...

– *Compare(TCM1_ TimeAtSea, ?pred, Month)* is a primitive formula. There is no answer since we can not evaluate the comparison of two durations both greater than a week *(TCM1_ TimeAtSea, >, Week)* and *(Month, >, Week)* with no more information:

$?pred$

– *Compare(Winter,?pred,CrabSeason)* is a primitive formula. This primitive formula asks the relations between the *Winter* and the *CrabSeason*. Since the *CrabSeason* starts at the end of the summer and ends at its beginning, the *Winter* is during the *CrabSeason* and thus shorter. We obtain the three following tuples:

$?pred$
$during$
$Inside$
$<$

Although the complexity of the inferences is high (at least EXPTIME), it has not been a problem in our system for two reasons. Firstly, a cognitive map is hand designed and it is a visual model so it is usually quite a small graph, for instance in the Kifanlo project a thirty nodes map is a big one. Secondly, the saturated set is precomputed and queries give an answer in an acceptable time in our application. Nevertheless, to go one step further, a study should be done to evaluate the theoretical complexity and how to face it depending on maps structure.

6 Temporal Cognitive Map Query Language

TCMQL is the extension of CMQL that integrates the temporal primitives *TimeInfo* and *Compare*. TCMQL is designed to query a set of temporal cognitive maps defined on the same temporal ontology.

TCMQL's syntax is close to SQL's syntax : `SELECT` selects variables `?x`..., `FROM` indicates the maps to query and `WHERE` describes the conditions. Four examples are given here along with their results and comments, they are based on $\mathcal{T}_1$ (Fig. 1), $\mathcal{O}_1$ (Fig. 5) and $TCM1, TCM2$ (Fig. 6).

Our aim is to provide a simple and intuitive language: a user does not need to precisely understand the underlying semantics. In this section we first describe examples of TCMQL queries and their result, then we provide syntax and semantics.

6.1 Queries

Example 18. The primitives `IsInMap` and `KindOf` are used in this example. In plain English this query means: 'In which maps are used the concepts types of Pleasure?'.

```
SELECT ?map,?concept FROM TCM1,TCM2 WHERE{
KindOf(?concept, Pleasure)
AND IsInMap(?map,?concept)
}
```

The first condition allows to obtain the concepts that are types of Pleasure in the taxonomy. The second condition gets the couples (map, concept) such that the concept belongs to the map. The result of this query is the list of the following tuples (?map,?concept):

?map	*?concept*
TCM1	*BossPleasure*
TCM1	*CrewPleasure*
TCM2	*BossPleasure*

The result shows what are the types of the concept pleasure and in which maps they appear.

Example 19. In plain English this query means: 'When does fisherman1(TCM1) fish crabs in comparison to fisherman2(TCM2)?'.

```
SELECT ?pred FROM TCM1,TCM2 WHERE{
TimeInfo(TCM1, CrabFishing, ?e1) AND
TimeInfo(TCM2, CrabFishing, ?e2) AND
Compare(?e1,?pred,?e2) }
```

The first two conditions allow to get the temporal entities of the concept CrabFishing in TCM1 and TCM2. The third condition allows to get all comparison predicates between those two temporal entities that are characterized by "finishes CrabSeason" and "= Month" for the one in TCM1 and by "equals CrabSeason" for the other. The result is made of the tuples (?pred):

?pred
finishes
$<$

The result shows that the fisherman1 fishes at the end of the fisherman2's fishing period, for a shorter period.

Example 20. In plain English this query means: 'Which duration of FishingSets influences BossPleasure?'.

```
SELECT ?p, ?e2, ?map FROM TCM1,TCM2 WHERE{
Path(?map,FishingSet,BossPleasure,?path)
AND TimeInfo(?map,FishingSet,?e1)
AND Compare(?e1, ?p, ?e2))}
```

The first condition allows to get the maps in which FishingSet influences BossPleasure (TCM2). The two following conditions allow to get the temporal information about FishingSet in the right map.

$?p$	$?e2$	$?map$
$=$	$ShortFishingSet$	$TCM2$
$<$	Day	$TCM2$
...	...	...

The result shows that according to the fisherman2 (TCM2), the BossPleasure is influenced by a short period of fishingset.

Example 21. This query asks the concepts in summer which influence a concept kind of `Pleasure`.

```
SELECT ?map,?c1,?i,?c2 FROM TCM1,TCM2
WHERE{KindOf(?c2,Pleasure) AND
Value(?map,?c1,?c2,?i) AND ?i != 0 AND
TimeInfo(?map,?c1,?e1) AND
Compare(?e1,in,Summer)}
```

The first condition allows to get all concepts ?c2 kind of Pleasure. The second and third ones allow to get the concepts ?c1 that influences ?c2 with their influence value. The two last conditions filter only the concepts ?c1 in summer.

$?map$	$?c1$	$?i$	$?c2$
$TCM1$	$OffSeason$	-0.5	$BossPleasure$
$TCM1$	$OffSeason$	0.5	$CrewPleasure$

The results show that, according to the fisherman1, the OffSeason which is in Summer influences negatively the pleasure of the boss and positively the pleasure of the crew.

6.2 Syntax and Semantics

Generally speaking, a query is mainly composed of variables to be returned, a data source and formulas. These formulas are recursive and can be combined with each other according to different rules. The atomic formulas are either primitive formulas or expression formulas. The primitive formulas, presented in the previous section, are composed of a primitive name and terms being either variables or constants. Expression formulas are composed of two terms and of an operator. We present here a (simplified) syntax of the TCMQL language in the Backus-Naur form (BNF).

Query ::= SelectClause FromClause WhereClause
SelectClause ::= '`SELECT`' ResultsClause ('`,`' ResultClause)*
ResultClause ::= '`DISTINCT`'? Variable
FromClause ::= '`FROM`' ('`ALL`'| *MapName* ('`,`' *MapName*)*)
WhereClause ::= '`WHERE`' "FormulaClause"
FormulaClause ::= '`(`' FormulaClause '`)`' (1)
| FormulaClause '`AND`' FormulaClause (2)
| FormulaClause '`OR`' FormulaClause (3)
| AtomicClause
AtomicClause ::= ExpressionClause | PrimitiveClause
ExpressionClause ::= T Operator T (4)
Operator ::= '`<`' | '`<=`' | '`=`' | '`!=`' | '`>=`' | '`>`'
T ::= Variable | *Constant*
Variable ::= '`?`' *VariableName*
PrimitiveClause ::= *PrimitiveName* '`(`' T ('`,`' T) + '`)`' (5)

The semantics of TCMQL is based on the semantics of the domain relational calculus. We describe here an partial and simplified version of the semantics. The main idea behind this definition of semantics is that each numbered formula above denotes a relation whose attributes are the variables of the formula. We need to introduce some notations first:

- Let F be a formula and let $v(F)$ be the set of variables of F, each variable v_i has an associated domain $d(v_i)$.
- Let V be a set of variables, $d(V)$ is the set of associated domains.
- Let $V = \{V_1, \ldots, V_n\}$ be a set of variables, $\Pi(V : d(V))$ denotes the cartesian product of domains $d(V_1) \times \ldots \times d(V_n)$ indexed by the elements of V.

Let F be a formula, the meaning of F, denoted $mng(F)$, is a relation of arity $|v(F)|$ such that:

(1) $mng((F)) = mng(F)$
(2) $mng(F_1 \text{ AND } F_2) = mng(F_1) \times \Pi(v(F_2) - v(F_1) : d(v(F_2) - v(F_1))) \bigcap mng(F_2) \times \Pi(v(F_1) - v(F_2) : d(v(F_1) - v(F_1)))$
(3) $mng(F_1 \text{ OR } F_2) = mng(F_1) \times \Pi(v(F_2) - v(F_1) : d(v(F_2) - v(F_1))) \bigcup mng(F_2) \times \Pi(v(F_1) - v(F_2) : d(v(F_1) - v(F_1)))$
(4) $mng(T_1 \text{ op } T_2) =$
  - $\{t \in \Pi(T_1 : d(T_1))/t(T_1) \text{ op } T_2\}$ if T_1 is a variable and T_2 is a constant.
  - $\{t \in \Pi(T_2 : d(T_2))/T_1 \text{ op } t(T_2)\}$ if T_1 is a constant and T_2 is a variable.
  - $\{t \in \Pi(\{T_1, T_2\} : d(\{T_1, T_2\}))/t(T_1) \text{ op } t(T_2)\}$ if both T_1 and T_2 are variables.

(5) $mng(\text{PrimitiveName}(T_1, \ldots, T_n))$ = value of the primitive formula (cf. definition 12)

7 Conclusion

This paper introduces a new model of cognitive map, called temporal cognitive map, that allows to extend cognitive maps with temporal features, using an ontology which includes periodic intervals. This paper also proposes a language, called TCMQL, which allows to query sets of temporal cognitive maps.

The temporal features of temporal cognitive maps are issued of real needs. The ACS project uses temporal cognitive maps to better modelling fishermen's strategies that were expressed in the Kifanlo project with simple cognitive maps. The VSPCC software has been developed to edit and use temporal cognitive maps. VSPCC can also execute TCMQL queries. The implementation uses the temporal ontology owl-time to which is added a class PeriodicInterval as a subclass of the main class TemporalEntity and comparison predicates as properties. VSPCC is available online [19].

Two perspectives of the research can be given. First, we are currently working with geographers of the LETG laboratory (University of Nantes, CNRS UMR(6554). TCMQL is used to analyze the fishing strategy in Atlantic from 1970 to nowadays. These results will be published in a geographical journal. The promotion of TCMQL in international laboratories of geography is one of our aims. Second, in our computer science laboratory, evolution of TCMQL should be considered in the next years. The merge of temporal aspects in TCMQL with geographical aspects is in progress. A reflection on how to compare automatically maps to provide learning abilities to our system has to be taken.

References

1. Axelrod, R.M.: Structure of Decision: The Cognitive Maps of Political Elites. Princeton University Press, Princeton (1976)
2. Balbiani, P., Osmani, A.: A model for reasoning about topologic relations between cyclic intervals. In: Principles of Knowledge Representation and Reasoning, pp. 378–385 (2000)
3. Carvalho, J.P., Tome, J.A.B.: Rule based fuzzy cognitive maps-expressing time in qualitative system dynamics. In: 10th IEEE International Conference on Fuzzy Systems, vol. 1, pp. 280–283. IEEE (2001)
4. Çelik, F.D., Ozesmi, U., Akdogan, A.: Participatory ecosystem management planning at Tuzla lake (Turkey) using fuzzy cognitive mapping. arXiv preprint q-bio/0510015 (2005)
5. Chauvin, L., Genest, D., Loiseau, S.: Ontological cognitive map. Int. J. Artif. Intell. Tools **18**(05), 697–716 (2009)
6. Codd, E.F.: The Relational Model for Database Management: Version 2. Addison-Wesley Longman Publishing Co. Inc., Boston (1990)
7. Ermolayev, V., Batsakis, S., Keberle, N., Tatarintseva, O., Antoniou, G.: Ontologies of time: review and trends. Int. J. Comput. Sci. Appl. **11**(3), 57–115 (2014)
8. Ermolayev, V., Keberle, N., Matzke, W.-E., Sohnius, R.: Fuzzy time intervals for simulating actions. In: Kaschek, R., Kop, C., Steinberger, C., Fliedl, G. (eds.) UNISCON 2008. LNBIP, vol. 5, pp. 429–444. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78942-0_42

9. Francis, N., et al.: Cypher: an evolving query language for property graphs. In: Proceedings of the 2018 International Conference on Management of Data, SIGMOD 2018, pp. 1433–1445. Association for Computing Machinery (2018)
10. Genest, D., Loiseau, S.: Modélisation, classification et propagation dans des réseaux d'influence. Technique et Science Informatiques **26**, 471–496 (2007)
11. Harris, S., Seaborne, A., Prud'Hommeaux, E.: SPARQL 1.1 Query Language. W3C recommendation (2013)
12. Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P.F., Rudolph, S.: OWL 2 web ontology language primer. W3C recommendation (2009)
13. Hobbs, J.R., Pan, F.: Time ontology in OWL. W3C Candidate Recommendation (2020). https://www.w3.org/TR/owl-time/
14. International Organization for Standardization: ISO IEC 9075:2016: Information Technology - Database languages SQL. ISO, Geneva, Switzerland (2016)
15. Kosko, B.: Fuzzy cognitive maps. Int. J. Man Mach. Stud. **24**, 65–75 (1986)
16. Ladkin, P.B.: Time representation: a taxonomy of internal relations. In: AAAI, pp. 360–366 (1986)
17. Le Dorze, A.: Validation, synthèse et paramétrage des cartes cognitives. Ph.D. thesis, LERIA, Université d'Angers, France, November 2013
18. Le Dorze, A., Chauvin, L., Garcia, L., Genest, D., Loiseau, S.: Views and synthesis of cognitive maps. In: Ramsay, A., Agre, G. (eds.) AIMSA 2012. LNCS (LNAI), vol. 7557, pp. 119–124. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33185-5_13
19. LeDorze, A., Robert, A.: https://sourcesup.renater.fr/projects/vspcc (2016)
20. Louis, G., Pirotte, A.: A denotational definition of the semantics of DRC, a domain relational calculus. In: VLDB, pp. 348–356 (1982)
21. Martin, B.L., Mintzes, J.J., Clavijo, I.E.: Restructuring knowledge in biology: cognitive processes and metacognitive reflections. Int. J. Sci. Educ. **22**(3), 303–323 (2000)
22. Osmani, A.: Introduction to reasoning about cyclic intervals. In: Imam, I., Kodratoff, Y., El-Dessouki, A., Ali, M. (eds.) IEA/AIE 1999. LNCS (LNAI), vol. 1611, pp. 698–706. Springer, Heidelberg (1999). https://doi.org/10.1007/978-3-540-48765-4_74
23. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Elsevier, Amsterdam (2014)
24. Robert, A.: (2019). http://www.info.univ-angers.fr/pub/adrian/rules/rules.html
25. Robert, A., Genest, D., Loiseau, S.: A query language for cognitive maps. In: Agre, G., van Genabith, J., Declerck, T. (eds.) AIMSA 2018. LNCS (LNAI), vol. 11089, pp. 218–227. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99344-7_20
26. Robert, A., Genest, D., Loiseau, S.: The taxonomic cognitive map query language: a general approach to analyse cognitive maps. In: 30th ICTAI, pp. 999–1006. IEEE (2019)
27. Robert, A., Genest, D., Loiseau, S.: Temporal cognitive maps. In: Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART 2020), vol. 2, pp. 58–68. INSTICC, SciTePress (2020)
28. Röhrig, R.: A theory for qualitative spatial reasoning based on order relations. In: Proceedings of the Twelfth AAAI National Conference on Artificial Intelligence, AAAI 1994, pp. 1418–1423. AAAI Press (1994)
29. Zhong, H., Miao, C., Shen, Z., Feng, Y.: Temporal fuzzy cognitive maps. In: 2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence), pp. 1831–1840. IEEE (2008)