Wydział Matematyki i Informatyki
Uniwersytetu im. Adama Mickiewicza w Poznaniu

Środowiskowe Studia Doktoranckie
z Nauk Matematycznych

# Extracting, Annotating and Reasoning about Time and Space in Texts and Discourse

Gérard Ligozat

Paris Sud University
gligozat@gmail.coml

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI
UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Publikacja współfinansowana ze środków Uni Europejskiej
w ramach Europejskiego Funduszu Społecznego

## 1. Introduction

Understanding the temporal structure of a text is a significant component of text understanding: a reader has to understand the **temporal relations** (e.g. precedence, inclusion) between events and time-denoting expressions in the text.

Prior research in **linguistics** has explored the various knowledge sources that are involved in the understanding of temporal notions. These sources include tense, aspect, temporal adverbials, rhetorical relations, pragmatic conventions, and background knowledge.

Recovering the temporal structure of documents has important **applications**. For instance, in multidocument summarization, determining the relative order of events is implied in merging them in a significant way; in question answering, it may be necessary to locate events on the time line, or to determine the events occurring prior to some event or date.

The research community has now developed a strategy for addressing the problem. It starts with the use of **annotation schemes**. They are used to tag the elements in a text which have a temporal significance. TimeML is the most widely used annotation scheme for temporal annotation. An annotated text is a point of departure for a subsequent processing of the temporal information.

**Temporal reasoning** is involved in at least two ways. Firstly, a text annotated by a human annotator only contains information about a small subset of all possible pairs of events. A human reader will be content of this sparse information, and will only use further reasoning on demand. But this implies that no guarantee of consistency is provided for the annotated text. Secondly, this sparseness of temporal data is a drawback for using machine learning methods.

After describing the linguistic background of temporal information understanding, we will describe how texts are annotated. For the sake of completeness, we describe briefly the similar problem of the tagging of spatial information.

**Qualitative spatial and temporal reasoning** has now developed into a significant sub-domain in AI. As mentioned above, it plays a significant role in at least two phases of the process of understanding time and space in texts. We will describe the main calculi which have developed in the past decades after Allen's pioneering work. We also mention their extensions to metrical notions (called **hybrid calculi**), and to the **fuzzy** framework. Metrical notions may be useful in representing quantitative notions in time (e.g. durations), while fuzzy notions are one way to address vagueness in time or space.

Ultimately, time and space will often have to be considered simultaneously when dynamics and motion are involved. Spatio-temporal annotation schemes are currently being developed. Qualitative **spatio-temporal** reasoning calculi are also an active domain of investigation. Exploring how both topics relate and cooperate for understanding spatio-temporal knowledge in a text is a promising perspective for future research.

## 2. Time and space in natural language

The temporal information in natural language is conveyed by **temporal markers**. The traditional grammatical terminology is ambiguous, since it uses for instance the term **tense** to refer to a grammatical category (simple past, pluperfect are two verbal tenses of English), but also to a semantic notion, the location of an event in time.

### 2.1. Temporal markers in natural language

**Verbs** are the main carriers of temporality in languages such as English. A verb may be tensed: *I visited* or not: *to visit*, *visiting*.

– **Tense** is grammaticalized by markers on the verb (*I stayed* using the marker *-ed*) or on the verb phrase (*I will visit*).
– **Aspect** in English is concerned with the opposition of *simple* forms such as *I stayed* and **progressive** forms such as *I was staying*

**Nouns** can also refer to events located in time, either because they may be considered as nominalizations: *my only visit*, or because they refer to temporal entities, either intrinsically: *that day*, or contextually *the party*.
**Adjectives** such as *recent*, *former*, *current*, *daily*, *biannual*, *long-term*, have a contribution to the temporal interpretation of the texts they belong to.
**Temporal adverbials** constitute another category of linguistic markers for temporal information: *at that time*, *one evening*, *after my arrival*.
**Clauses** such as *when I arrived*, *when we played tennis together* also have to be considered.

### 2.2. Semantics of tense and aspect

**T**he semantic category of **tense** is concerned with the location of events in time. For instance, *I visited the castle* conveys the information that the visit took place before the so-called **time of speech**. This information can be carried by the verbal tense, but it can also be indicated by temporal adverbial or temporal clauses, as well as relative to a calendar (*September 11*), nouns or noun phrases (*the day after my arrival*).

**Aspect** in English is mainly associated to the verbal markers *-ing* (progressive aspect) which is a form of **imperfective**. The distinction between the imperfective form in *Peter was crossing the street* and the **perfective** form *Peter crossed the street* may be crucial for correct understanding : the first sentence is consistent with *He never reached the sidewalk*, while the second one is not.

Languages exhibit quite different linguistic markers of aspect. French has a simple past tense (perfective) which is in opposition to **imparfait**, an imperfective tense. Slavic languages have **aspectual pairs** of verbs (in first approximation, such that *pisat'/napisat'* (to write, imperfective/ to write, perfective). Languages which have no verbal tenses, such as Mandarin Chinese, use verbal particles to indicate aspectual notions. Some languages have temporal markers on nouns. Notice that the *ex-* morpheme in English is a sort of temporal marker which allows to derive *ex-wife* from *wife*.

According to Comrie [9], the semantic category of aspect may be structured as follows:

For imperfective and perfective aspects:

- **Imperfective**
  - Habitual: *Twenty years ago, we used floppy disks.*
  - Continuous
    - Non-progressive: *When I visit John, he'll recite his poems.*
    - Progressive: *When I visit John, he'll be reciting his poems.*
- **Perfective**: *The bomb exploded at 3am.*

For Perfect:

- Result: *The file has been saved.*
- Experiential: *Have you ever eaten snails?*
- Persistent situation: *Since 2000, peter has lived in Warsaw.*
- Recent past: *I have just finished my letter.*

#### 2.2.1. Aspects in Polish

We will illustrate the phenomenon of aspect in a Slavic language by using the example of Polish. Our presentation will be based on Młynarczyk's thesis [36].

1. Perfective verbs may be formed from imperfective ones by prefixation: *pisać* (I) / *napisać* (P) (to write).
2. Imperfective verbs may be formed from perfective ones by suffixation: *kupić* (P) / *kupować* (I) (to buy).
3. Some aspectual pairs are **suppletive pairs** which are morphologically unrelated, such as *brać* (I) / *wziąć* (P) (to take).

Moreover, while the "neutral" perfective form corresponding to *pisać* (I) is indeed *napisać*, other perfective verbs are derived from *pisać* (I):

- *popisać* (to write);
- *podpisać* (to sign);
- *przepisać* (to copy);
- *przepisać* (to prescribe);
- *zapisać* (to write down).

Prefixes which do not change the meaning of the imperfective form are called **empty prefixes.**

Młynarczyk proposes a classification of aspectual pairs in Polish into five classes according to the morphological processes they can enter:

1. Pairs formed by adding an empty prefix;
2. Pairs formed by adding the **delimitative** prefix *po-*;
3. Pairs formed by inserting the **semelfactive** (instantaneous) suffix *-ną*;
4. Pairs making use of suffixation possibly combined with vowel change.

The five classes are defined as summarized in Table 1.

This formal classification is claimed to be related to the semantic classification of those verbs in terms of modes of process as described below.

| | ep | po- | -ną | mpc |
|---|---|---|---|---|
| class1 | yes | | | |
| class2 | | yes | | |
| class3 | yes | yes | | |
| class4 | yes | yes | yes | |
| class5 | | | | yes |

Table 1. Młynarczyk's classification of Polish aspectual pairs

### 2.3. Spatial information in natural language

#### 2.3.1. Spatial markers

Linguistic markers of location comprise proper names, nouns and noun phrases, prepositions, and verbs.

- **Proper names** like *Warsaw*, *New York*, *Paris*, which refer to places.
- **Nouns** and nominals, like *city*, *town*, *the Forbidden City*, *the Bulgarian capital* which also refer to places.
- **Prepositions** may be directional, like *north*, *south-east*, topological like *in the Alps*, *across the border*; they can be qualified by distances: *100 miles north.* They refer to relations between locations.
- **Verbs** comprise verbs of location such as *to lie*, *to be situated*, *to extend*, and verbs of motion: *to enter*, *to cross*, *to reach*, *to avoid.*

#### 2.3.2. Semantics of spatial expressions

The study of the semantics of spatial markers is a notoriously difficult field of research in linguistics. We will only mention here some facts related to this topic:

- Places can be underspecified, and only unambiguously defined contextually: *the church*, *the Mall.*
- Proper names for places (toponyms) can be ambiguous: *Cambridge* (*Cambridge, UK*, or *Cambridge, MA*?), *Paris* (*Paris, France*, or *Paris, Texas*?).
- In many cases, locating a spatial entity depends on using the right **frame of reference**: *in front of the church*, *behind the car*, *in front of you.*
- The semantics of prepositions cannot be reduced to topological or geometrical notions only: *in the cup*, *on the ceiling.* At least the **functional** characteristics of the objects have to be taken into account: you drink out of a cup, so that a lump of sugar is *under* a cup, not *in* it if the cup is upside down; a fly is supported by the ceiling when it crawls *upon* it.
- The semantics of motion verbs is quite complex: *to cross*, *to avoid* are cases in point.

## 3. Formalizing temporal notions

The semantics of tense and aspect has been a topic of study for linguists and philosophers at least since Aristotle, who remarked that the Greek equivalents of *Socrates crossed the street* and *Socrates was crossing the street* have different inferential properties:

the second sentence does not imply that *Socrates has crossed the street*, whereas the first one does.

Reichenbach [50] proposed to account for the fact that English has more than one "past" tense by using three temporal indexes: S ("time of speech"), R ("time of reference"), and E ("time of event"). Each index can precede, follow, or coincide with another. Then, each grammatical tense of English corresponds to a particular configuration. For instance, $S = R = E$ corresponds to the *present tense*, $E < R < S$ the *plus quam perfect*, $S < E < R$ the *future anterior*.

This description of morphological tense has been extended in various ways (for instance, intervals rather than point may be used as indexes). Alternatively, Bruce [7] has a system where an arbitrary number of indexes may be used.

Locating indexes of reference is usually accomplished using temporal adverbials. Temporal adverbials are of two types: deictic or non-deictic. By definition, deictic elements can only be interpreted relatively to the coordinates of the speaker. For time, *now*, *yesterday*, *next month* are deictic adverbials, because their interpretation is relative to the time of utterance, whereas *at that time*, *the day after*, depend on contextual elements for their interpretation.

Real world processes possess characteristics such as being static or dynamic, having an initial or a terminal phase, being punctual or durative, etc. The *mode of process* is the linguistic realization of these characteristics. In the same way as grammatical tense is not congruent with time, the mode of process cannot be identified with the physical reality of the process; rather, it is a conceptual representation of this process.

A widely use classification of modes of process is Vendler's classification. According to Vendler, verbs (or verb phrases) can be assigned to one of the following four classes:

- **States** are processes which have neither starting point nor finishing point (in their linguistic conceptualization), and which are not compatible with a punctual description; for instance: *Watson* knows *how to lead an investigation.*
- **Activities** are processes without a determined end point, which are durative and not conclusive: *Watson* is investigating *the Densmore case.*
- **Accomplishments**, as opposed to activities, are durative dynamic processes which are conclusive: *Watson* is solving *the Densmore case.*
- **Achievements** are processes which are conceived as punctual: *I stopped smoking.*

Semantic tests can be applied to distinguish between the four classes. For example, achievements with an indication of duration have to be interpreted as repeated events: *Watson has been stopping smoking for three years.*

It is commonly assumed that this classification is concerned with verb phrases rather than verbs themselves. In particular, the mode of process can depend on the fact that the direct complement of a verb is defined or not. Compare:

1. *Watson* solved *the Densmore case.*
2. *Watson* solves *cases.*

In (1), the verb phrase is an accomplishment, whereas in (2), it is an activity.

In their analysis of the semantics of temporal expressions, Moens and Steedmann [37] consider the following example:

"*When they built the 39th Street b r i d g e . . . ,*

1. *. . . a local architect drew up the plans*;
2. *. . . they used the best materials*;
3. *. . . they solved most of their traffic problems*".

What are the mechanisms which help us to understand that 1) refers to the preparatory period of the construction, 2) refers to the period of construction itself, and 3) refers to the state of affairs resulting from the completion of the construction? To answer this question, the authors propose to analyze each event in terms of entities which occupy two contiguous temporal intervals: a *preparatory process*, followed by a *culmination point*, which in turn is followed by a *resulting state* (Figure 3.2).

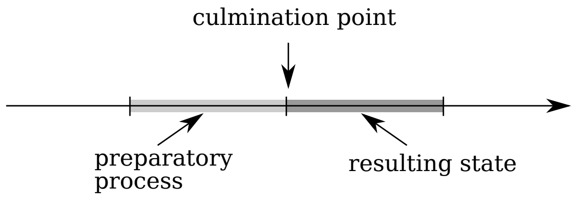


Figure 1. A sequence containing a preparatory process, a culmination point, and a resulting state is associated with each event

Thus, a temporal entity is associated with an event and this temporal entity comprises not two (its beginning and its end) but three successive points, since a culmination point is added between the first two.

### 3.1. Relations between events

Determining the relations between events and representing those relations will be discussed below. We will only mention here the fact that the language defined by Allen [1] for describing the relations between two intervals is widely used. It consists of a qualitative language based on the relative positions of the endpoints of the intervals. Figure 2 represents the set of relations, called **basic relations** of the language.

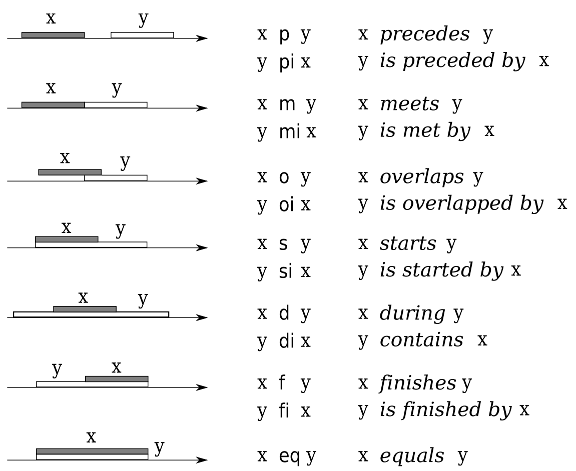


Figure 2. Allen's basic relations between two intervals

We will see below that the annotation schemes for representing temporal information use a subset of Allen's relations.

## 4. Extracting temporal and spatial information

Linguistic annotation can be done by hand, but it is a slow and tedious process. Software tools have been devised by the research community to make things easier. Moreover, different user annotators usually produce different annotations. For those texts and corpora with an agreed annotation by humans, one uses the term of **gold standard** annotations.

The development of the statistical approach in linguistics has led to the elaboration of statistical models for doing automatic annotation. These models use machine learning (ML) methods. Typically, an already annotated corpus is divided into a **training corpus** and a **test** corpus. The system is trained on the training corpus, then tested on the test corpus.

### 4.1. Annotating texts

Linguistic annotations are notes about features that are attached to some objects in a text. For example, a **part-of-speech tagger** adds an annotation to a word in the text to say that this word is a noun, a verb, an adjective, a preposition, and so on.

Annotation has now become a pervasive technique for extracting information from a text. For example, consider a speech synthesizer which has to read the word *record*. The system has to know whether this word is a noun or a verb in order to pronounce it correctly.

**Inline annotation** modifies the original text by inserting the annotations in it. By contrast, **standoff annotation** uses a separate file that contains the annotations and pointers that reference the original document.

One distinguishes between several levels of annotation. For instance:

- **Sentence boundary recognition**, which segments the text into sentences.
- **Tokenization** segments it into words (tokens).
- **Part-of-speech tagging** (POS-tagging) tags word with word category tags.
- **Chunk tagging** segments it into chunks, which are word sequences that have a syntactic consistency (typically, noun phrases), while other sequences are left untagged.
- **Syntactic parsing** identifies syntactic constituency relationships between phrases in a sentence (phrase-structure grammars), or alternatively dependency relationships between words.
- **Semantic analysis** identifies the semantic components of a sentence and the relations between them, for instance in terms of a predicate-argument structure.
- **Named-entity recognition** (NER) identifies and tags named entities, that is definite noun phrases which refer to specific types of individuals, such as organizations, persons, dates, countries, and so on.
- **Co-reference resolution** links references (typically pronouns such as *she*, *he*,*it*, *they*) to the entity in the text they refer to.

For the most frequent types of annotations, such as POS-tagging, several standards are being used by the research community. A typical instance is the Penn Treebank tagset, which uses 47 tags, including six different tags for verbs: VB, VBZ, VBP, VBD, VBN, VBG, illustrated by *be*, *walks*, *is*, *walked*, *given*, *walking*, respectively.

#### 4.1.1. Annotating the temporal and spatial content

Annotating a text for its temporal content implies tagging the markers which refer to temporal entities, and identifying and tagging the relations between them. Similarly for the spatial content, where markers for locations have to be tagged together with their relationships. This means that temporal and spatial tagging involve special cases of NER (recognizing events, dates, locations, temporal and spatial relations) as well as co-reference resolution.

### 4.2. Temporal and spatial annotation schemes

Informally, an annotation scheme comprises a set of tags, together with detailed and precise indications on the way to use them.

Following the work of Pustejovsky and other researchers [45], specific annotation schemes have been developed for temporal annotation. One popular such scheme is TimeML, which has now spawned a ISO standard called ISO-TimeML.

More recently, annotation schemes for spatial information have been proposed. We will briefly describe SpatialML [32], which is the spatial analog of TimeML.

As their names indicate, both schemes are based on XML.

#### 4.2.1. TimeML

TimeML is a specification language for events and temporal expressions in natural language. It is designed to address four problems in event and temporal expression markup:

- Time stamping of events (identifying an event and anchoring it in time).
- Ordering events with respect to one another (lexical versus discourse properties of ordering).
- Reasoning with contextually underspecified temporal expressions (temporal functions such as ’last week’ and ’two weeks before’).
- Reasoning about the persistence of events (how long an event or the outcome of an event lasts).

TimeML has been developed in the context of three AQUAINT workshops and projects. The 2002 TERQAS workshop set out to enhance natural language question answering systems to answer temporally-based questions about the events and entities in news articles. The first version of TimeML was defined and the TimeBank corpus was created as an illustration. TANGO was a follow-up workshop in which a graphical annotation tool was developed. Currently, the TARSQI project develops algorithms that tag events and time expressions in NL texts and temporally anchor and order the events.

The original TimeML specification has developed into more recent versions resulting in the ISO-TimeML language.

#### 4.2.2. TimeML in short

The set of TimeML tags contains the following elements for tagging “entities”:

– The ¡EVENT¿ tag is used to tag **events**; some of its attributes are `class`, which can take the values OCCURRENCE, IACTION, ISTATE, ASPECTUAL, PERCEPTION, REPORTING, and `aspect`, `modality`, `polarity`, `tense`, `vform`.
– The ¡TIMEX3¿ tag is used to tag **temporal expressions**, which can be dates, times, durations, or deictic expressions.
— The ¡SIGNAL¿ tag is used to tag **signals**, that is, relation markers (*during*, *before*, *after*, etc.
– The ¡TLINK¿ tag is used to tag **temporal links**, or temporal relations; these relation may be relations between two EVENTs, two TIMEX3s, or one EVENT to one TIMEX3; its attribute `relType` can have the values DURING, BEFORE, AFTER, etc. Its attributes `sid` and `tid` indicate the source and target of the relation.
– The ¡SLINK¿ tag is used to tag **subordination Links**, that is, links between a clause and a subordinate clause; its attribute `relType` can have the values INTENTIONAL, EVIDENTIAL, FACTITIVE, etc. For example, there is a SLINK between the two EVENTS associated to *assured* and *had stayed* in *she assured me that she had stayed.*
– The ¡ALINK¿ tag is used to tag **aspectual Links**; its attribute `relType` can have the values BEGINS, ENDS, CONTINUES, etc. For example, there is a SLINK between *stopped* and *asking* in *Watson stopped asking questions.*

Here is an example of an annotated sentence:

Watson ¡EVENT `id`="e1" `class`="OCCURRENCE"¿arrived¡/EVENT¿
¡SIGNAL `id`="S1"¿after¡/SIGNAL¿
¡TIMEX3 `id`="t1" `val`="T0900"¿ 9 a.m.¡/TIMEX3¿
¡TLINK `relType`="AFTER" `eid`="e1" `tid`="t1" `sid`="s1" /¿

We will elaborate later on the way TimeML is used to address the four questions mentioned above. Notice that a fundamental point is to have a *good* annotation guide for the language concerned. This guide has to make quite precise what has to be annotated, what *should not* be annotated, and how to annotate.

For the English language, according to Saurí *et al.* [52], the rate of agreement between annotators is 64% for nouns and 80% for verbs.

#### 4.2.3. SpatialML

The goal of the SpatialML annotation scheme was to emulate the progress made earlier on time expressions, as well as schemes for marking up events and linking them to times.

SpatialML can be related to the Geography Markup Language (GML) defined by the Open Geospatial Consortium (OGC), as well as Google Earth's Keyhole Markup Language (KML)8 to express geographical features.

#### 4.2.4. SpatialML in short

The following tags and attributes are defined in SpatialML:

– The ¡PLACE¿ tag is used for **places**; among its attributes are `county`, `state`, `country`, `continent`, `gazRef`, `latLong`, `form`, etc.

– The ¡RLINK¿ tag is used to express a spatial trajectory between a pair of locations; among its attributes are `source`, `target`, `direction`, `distance`, `frame`, `signals`, etc.
– The ¡LINK¿ tag is used to express containment, connection, or other topological relations between locations; among its attributes are `source`, `target` and `linkType`. etc.
– The ¡SIGNAL¿ tag is used for **Signals**; among its attributes are `distance` and `direction`. etc.

Each tag has an `id` attribute which uniquely identifies it. A `comment` attribute is also available for each tag.

Some comments about the non obvious attributes: `gazRef` is used to refer to a gazetteer, that is a document which specifies the geo-localizations of places; `latLong` contains the latitude and longitude coordinates; `form` refers to the nature of the textual marker: it can be a proper noun (NAM) or a nominal (NOM); `frame` refers to the frame-of-reference: viewer-based, intrinsic, or extrinsic; `linkType` refers to topological relations between regions which are a subset of the RCC-8 relations discussed later in this document.

Here are two examples of the use of SpatialML, which we adapt from the MITRE SpatialML document:

*Poznań, west of the Warsaw*
¡PLACE `type`="PPL" `id` =1 `form`="NAM"¿Poznań¡/PLACE¿
¡PLACE `type`="PPLC" `country`="PL" `id`=2 `form`="NAM"¿Warsaw¡/PLACE¿
¡SIGNAL `id`=3 `type`="DIRECTION"¿west¡/SIGNAL¿
¡RLINK `direction`="W" `source`=2 `signals`="3"/¿

*Kraków, Poland*
¡PLACE `type`="PPLC" `country`="PL" `id`=1 `form`="NAM"¿Kraków¡/PLACE¿
¡PLACE `type`="COUNTRY" `country`="PL" `id`=2 `form`="NAM"¿Poland¡/PLACE¿
¡LINK `source`=1 `target`=2 `linkType`="IN"/¿

The value PPL stand for "populated place", and "PPLC" for "capital city of a country".

## 5. Qualitative reasoning about time and space

### 5.1. Qualitative calculi

#### 5.1.1. The Point algebra

Qualitative calculi in Artificial Intelligence [30] aim to represent relations between entities which do not involve measurement. A typical instance is the Point calculus [63]: consider time points on the time line. In this calculus, we are only interested in the fact that, given two points, either they coincide, or one of them precedes the other.

In other words, if $U = \mathbb{R}$ is the time line, we consider three binary relations on it: denoting by $<$ the precedence relation, and by $>$ its inverse relation (which means that $x > y$ if and only if $y < x$), we consider the three relations $<$, $>$ and the identity relation `eq`. Hence the (infinite) set of all pairs of points is partitioned into three disjoint subsets corresponding to $<$, $>$, and `eq`.

**Relations** Even using this poor language to represent knowledge about points, we can still do some deduction. For instance, **inversion** tells us that, if $x < y$, then $y > x$, and conversely. Then **composition** of $<$ with itself assures us that, if $x < y$ and $y < z$, then $x <$, and in fact, conversely, if $x < z$, then there exists $y$ such that $x < y$ and $y < z$. Composing $<$ with $>$, by contrast, does not allow us to deduce anything new: if $x < y$ and $y > z$, then all three cases $x = z$, $x < z$, and $x > z$ are possible, and here again a converse property holds. This can be expressed as the fact that composing $<$ with $>$, or $>$ with $<$, results in the disjunction $U \times U$ of all three relations $<$, $>$, and `eq`, called basic relations.

Disjunctions of basic relations appear quite naturally as a way of expressing partial knowledge: for example, the union of $<$ and `eq` expresses that one point either precedes another, or is equal to it.

The set of all eight disjunctive relations, equipped with the set-theoretic operations, is a Boolean algebra; together with inversion and composition, it acquires an extra structure which makes it into an **relation algebra**. An element of this algebra is called a relation.

**Constraint networks** The eight relations may be used to represent knowledge about some finite set of points. For instance, Figure 3 represents what is called a **constraint network** on the Point algebra: the network has four nodes, which represent four points; the labels are elements of the Point algebra which assert that some relations hold between those points: for instance, point $V_2$ does not coincide with point $V_3$.

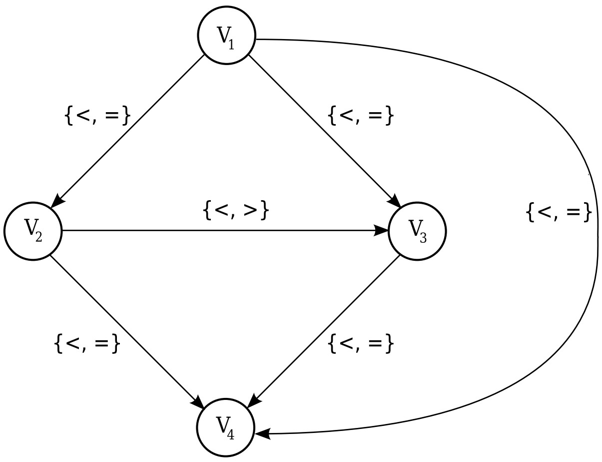


Figure 3. A constraint network on the Point algebra

Usually, a constraint network represents a set of constraints for which it it not known in advance whether they are consistent. The **consistency** problem consists in determining if there exists a set of entities — here, of points on the time line — satisfying all constraints. If this is the case, the configuration is called a solution of the network.

On the Point algebra, the consistency problem can be solved in polynomial time. This is no longer the case for Allen's algebra or for most of the qualitative temporal and spatial calculi, where it is a NP-complete problem.

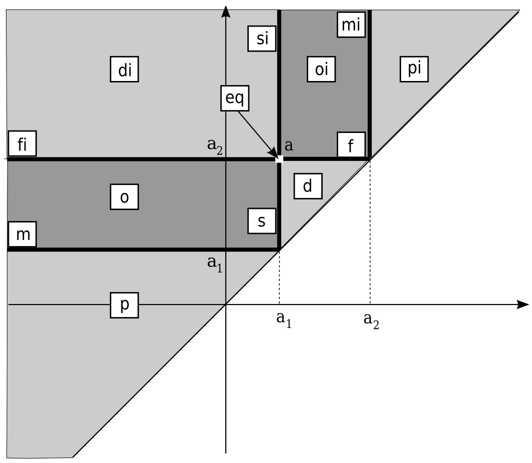


Figure 4. An interval and the 13 regions corresponding to Allen's basic relations

### 5.1.2. Allen's calculus

As mentioned before, Allen's calculus considers temporal entities which are (finite and bounded) intervals, characterized by their starting and finishing points: an interval is a pair $(x, y)$, where $x < y$. As a consequence, an interval can be represented by a point $(x, y)$ in the plane, with $x < y$, that is, a point which lies above the first bisector. Figure 4 represents such a point $(a_1, a_2)$.

Now, given an interval $(a_1, a_2)$, the intervals which are in one of the 13 basic relations with respect to it belong to a well-defined region in the plane, as represented in Figure 4.

Allen's algebra is the Boolean algebra of all subsets of the set of basic relations, so that it has 8192 elements. These disjunctive relation can be used to represent disjunctive knowledge about finite set of intervals, using constraint networks whose arcs are labeled by these relations.

**Complexity issues** The first main complexity result about Allen's calculus is the following:

**Theorem 1.** *For arbitrary constraint networks on Allen's algebra, the consistency problem is NP-complete.*

In the context of constraint networks on qualitative algebras, a favorite method for filtering out inconsistent network is the **algebraic closure** method. It is based on the following remark: for a 3-tuple $(i, j, k)$ of nodes, replacing the original constraint $C(i, j)$ on $(i, j)$ by its intersection with the composition of $C(i, k)$ by $C(k, j)$ does not change the problem.

Hence the method consists in repeating, for all 3-tuples $(i, j, k)$

$$C(i, j) \leftarrow C(i, j) \cap (C(i, k) \circ (k, j))$$

where $\circ$ denotes composition, until the network no longer changes. The resulting network is **algebraically closed** (or path-consistent): for any 3-tuple, one has:

$$C(i, j) \subseteq C(i, k) \circ (k, j)$$

If the empty relation appears during the calculation, then the network is consistent. But the converse is not true: there are algebraically closed networks on Allen's algebra which are not consistent. An example, due to Kautz according to Allen, is the network of Figure 5.

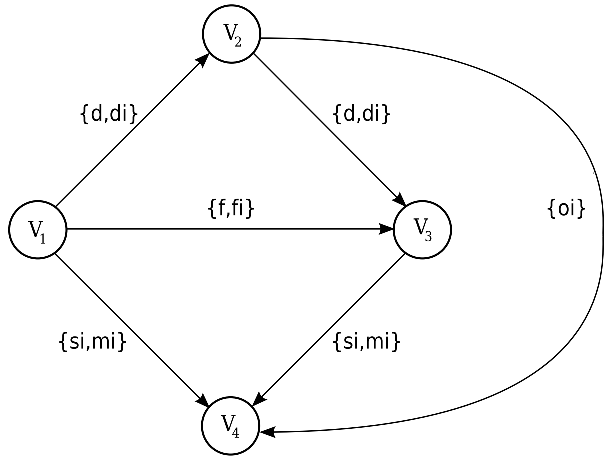


Figure 5. An algebraically closed inconsistent network (after Allen 1983)

Using suitable data structures, the algebraic closure algorithm can be implemented in $O(n^3)$ time, where $n$ is the number of nodes in the network.

**Looking for tractable sub-classes** In view of the intractability result for arbitrary relations, two related but distinct tasks are in order:

1. Identify subsets of relations for which the consistency problem is tractable.
2. Identify subsets of relations for which the algebraic closure method is a test for consistency.

Indeed, a sizable part of the research in the domain has been devoted to characterizing tractable sub-classes of the algebra. A first result in this direction is due to Vilain *et al.*. The relevant relations are the relations that can be described by a constraint network on their endpoints. These relations form a **sub-class**, which means that they are stable under inversion, intersection, and composition.

**Theorem 2.** *The consistency of networks whose relations are **pointizable** relations can be decided using the algebraic closure method.*

The final result on the subject was proved by Nebel and Bürckert in 2004 [43]:

**Theorem 3.** *There is a sub-class of relations in Allen's algebra which is the only maximal tractable sub-class containing all basic relations.*

**Proposition 1.** *The consistency of networks whose relations are **pre-convex** relations can be decided using the algebraic closure method.*

In Nebel and Bürckert's paper, those relations are characterized as ORD-Horn relations. Ligozat [28] gives another description of the sub-class in terms of **pre-convex** relations. This allows him to give visual test for recognizing whether a relation is tracta-

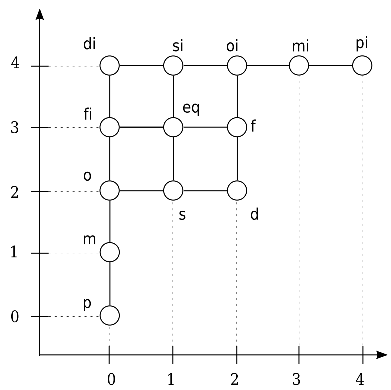


Figure 6. The lattice of basic Allen's relations

ble. It also allowed him to give an analytic proof of the maximality and uniqueness of the sub-class.

#### 5.1.3. A closer look at Allen's relations

Since each endpoint of an interval are points on the time line, they are provided with a natural ordering. Consequently, intervals, as pairs of points, are naturally ordered using the product ordering. This can be used for defining a partial order on basic relations. The resulting structure is a lattice called the **lattice of basic Allen's relations**. Figure 6 is a representation of this lattice.

Notice that dimensions can be assigned to relations by referring to their representation in terms of regions in the plane. Using the notion of dimension, we can now state:

**Proposition 2.** *In Allen's algebra:*
- Convex *relations are the relations that are* intervals *in the lattice.*
- Pointizable *relations are* odd-cutouts *of convex relations.*
- Pre-convex *relations are the relations that can be transformed into convex ones by adding basic relations of dimensions 0 or 1.*

### 5.2. Spatial and temporal calculi

Since the publication of Allen's paper in 1983, many qualitative calculi have been devised and studied by the qualitative reasoning community. They all fit a common schema: one considers a universe $U$ of temporal or spatial entities. Among all binary relations on $U$ — that is, among all subsets of $U \times U$, one chooses a finite family of (non-empty) relations $(R_i)_{i \in I}$ which is a partition of $U \times U$ (this is usually called the JEPD property, for "Jointly Exhaustive and Pairwise Disjoint"). If the family contains the identity relation, and if the inverse of each member of the family belongs to it, then the set up is called a **partition scheme** [29]. An operation of (weak) composition may then be defined, resulting in an algebra which in general is a non associative algebra, a notion which generalizes that of relation algebra.

Similar considerations may be defined for ternary, or higher arity relations.

Table 2 lists some of the most studied calculi of this type.

| Name of the calculus | Entities | Space | Relations | Arity | Absolute |
|---|---|---|---|---|---|
| Point | 1D points | $\mathbb{R}$ | 3 | binary | yes |
| Allen | 1D intervals | $\mathbb{R}$ | 13 | binary | yes |
| 1D directed point | 1D directed points | $\mathbb{R}$ | 6 | binary | no |
| Directed interval | 1D dipoles | $\mathbb{R}$ | 26 | binary | no |
| $\mathcal{OPRA}_m$ | 2D directed points | $\mathbb{R}^2$ | $4m(4m+1)$ | binary | no |
| 2D dipole | 2D dipoles | $\mathbb{R}^2$ | 16 | binary | no |
| Cardinal direction | points | $\mathbb{R}^2$ | 9 relations | binary | yes |
| Cardinal direction | regions | $\mathbb{R}^2$ | $218^1$ | binary | yes |
| Rectangle | rectangles | $\mathbb{R}^2$ | $13^2 = 169$ | binary | yes |
| $n$-point | points | $\mathbb{R}^n$ | $3^n$ | binary | yes |
| $n$-block | blocks | $\mathbb{R}^n$ | $13^n$ | binary | yes |
| INDU | intervals | $\mathbb{R}$ | 25 | binary | yes |
| $n$-star | points | $\mathbb{R}^2$ | $4n+1$ relations | binary | yes |
| Cyclic interval | dipoles | circle $S^1$ | 16 | binary | no |
| RCC-8 | regions | variable | 8 | binary | no |
| Discrete RCC-8 | regions | plane | 8 | binary | yes |
| Cyclic point | points | circle $S^1$ | 6 | ternary | no |
| Double cross | points | $\mathbb{R}^2$ | 17 | ternary | no |
| Qualitative triangulation | points | $\mathbb{R}^2$ | variable | ternary | no |
| 5-intersection | regions | plane/sphere | 34/42 | ternary | no |

Table 2. Qualitative spatial and temporal calculi

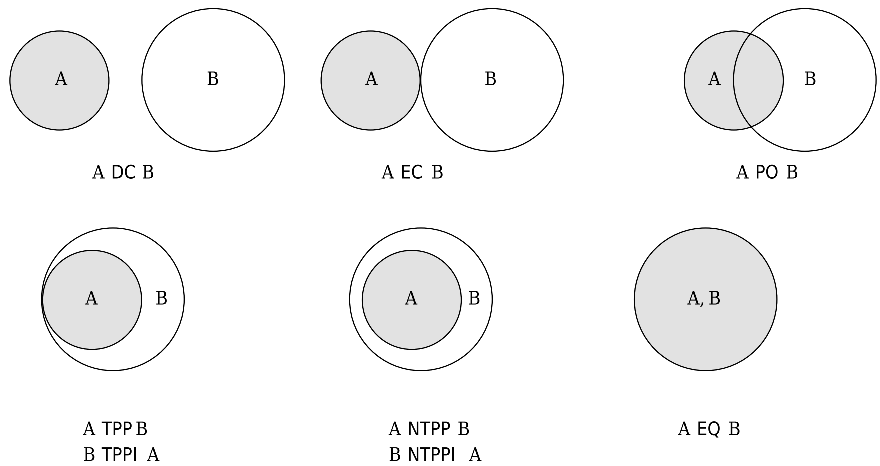


Figure 7. The basic relations of the RCC-8 calculus

### 5.3. The RCC-8 calculus

The RCC-8 calculus can be defined in a variety of ways. It originates in the work of Randell, Cui and Cohn [49] on the RCC logical theory for regions. One particularly simple way of defining the calculus is by using Egenhofer's model: consider all regions in the plane which are homeomorphic to a closed disk (intuitively, they are continuous deformations of a disk). Then partition the set of all pairs of regions into eight classes according to the fact that they belong to one of the situations illustrated by Figure 7. Then RCC-8 is the resulting calculus.

The calculus has eight basic relations — hence its name:

– DC for **disconnected**;
– EC for **externally connected**;
– PO for **partial overlap**;

– TPP for **tangential proper part** and its inverse TPPI;
– NTPP for **non-tangential proper part** and its inverse NTPPI;
– EQ for **equality**.

The calculus has many models which have been studied in detail ([57, 25]).

For our purpose here, we will only recall that it has been adopted by the SpatialML scheme for expressing topological relations between regions.

### 5.3.1. Tractability issues

The question of tractability was first studied by Nebel:

**Theorem 4.** *For constraint networks on the RCC-8 algebra:*
– *the consistency problem for the full algebra is NP-complete;*
– *it is tractable for atomic networks, and moreover the algebraic closure method decides consistency for those networks.*

Renz provides a complete determination of all tractable sub-classes of the RCC-8 algebra containing all basic relations:

**Theorem 5.** *In the RCC-8 algebra:*
– *there are three maximal tractable sub-classes of* RCC−8 *containing all basic relations, which are denoted by* $\hat{\mathcal{H}}_8$*,* $C_8$ *and* $Q_8$*;*
– *the algebraic closure method decides consistency for those sub-classes.*

This implies that computing "spatial closures" using the RCC-8 algebra is a tractable task provided that the disjunctive relations considered are restricted to one of the three tractable sub-classes.

### 5.3.2. Hybrid calculi

Two main approaches have been used to combine qualitative temporal reasoning with quantitative reasoning. The first approach is that of Meiri [34, 35], which represents quantitative constraints (between points) and qualitative constraints (between intervals) in the same constraint network. By contrast, the second approach, by Kautz and Ladkin [23], uses two distinct networks, one qualitative, the other quantitative, and defines ways of managing the interactions of the two types of constraints. We shall briefly present Meiri's approach.

Consider the following example taken from Meiri's paper [35]:

"John and Fred work for a company that has local and main offices in Los Angeles. They usually work at the local office, in which case it takes John less than 20 minutes and Fred 15-20 minutes to get to work. Twice a week John works at the main office, in which case his commute to work takes at least 60 minutes. Today John left home between 7:05-7:10 a.m., and Fred arrived at work between 7:50-7:55 a.m."

Assume further that John and Fred met on their way to work. Then Meiri's example is represented by the **hybrid** network of Figure 8, which involves both point-to-point constraints and Allen relations.

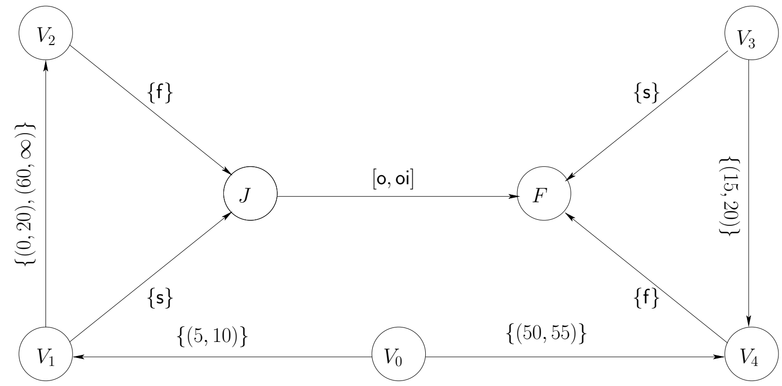


Figure 8. An hybrid network in Meiri's formalism

This network has seven nodes. Five of them correspond to time points, and two represent the intervals $J$ and $F$, corresponding to the respective journeys of John and Fred.

Constraints between time points are expressed using the language of TCSPs (Temporal Constraint Satisfaction Problems) of Dechter *et al.* [15] we indicate that $V_1$ starts: a label $I$ on the arc $(t, t')$, where $I$ is an interval or a union of intervals of the real line, means that the two time points satisfy the constraint $(t' - t =\in I$.

Constraints between a point and an interval are expressed using the relations s (starts) and f (finishes). For instance, $V_2$ finishes $J$, while $V_3$ starts $F$ and $V_4$ finishes it.

Finally, constraint between intervals are expressed in terms of Allen relations. Here, we state that intervals $J$ and $F$ have a common sub-interval.

Meiri's formalism generalizes Allen's calculus, the Point calculus and the TCSP formalism at the same time.

In the case of the Point calculus, the consistency problem is polynomial: the consistency of a network with $n$ vertices can be determined in $O(n^2)$ time. For Allen's calculus and for the TCSP formalism, this problem is NP-complete.

The qualitative Point-and-interval calculus is also NP-complete (the fact that it is NP-difficult results from a reduction of the *betweenness problem*.

Thus, Meiri's formalism itself is NP-complete, and Meiri poses the question of determining "islands of tractability" inside this formalism. Such a fragment is constituted by the Point calculus. In the case of Allen's algebra, the single tractable maximal subclass containing the basic relations is that of pre-convex relations. For the Point-and-interval calculus, the geometrical approach in terms of pre-convex relations provides a tractable subclass of relations:

**Theorem 6.** *In the algebra of qualitative relations between points and intervals, the subclass formed by pre-convex relations is polynomial [27]. Any pre-convex and non-zero algebraically closed network is consistent.*

### 5.3.3. Fuzzy calculi

The language of Allen's algebra allows us to express a certain type of uncertainty with the help of disjunction of basic relations. For example, we may express the fact that the periods of two meetings $I$ and $J$ are disjoint by enforcing the constraint $\{\mathsf{p}, \mathsf{pi}\}$ between them.

But we cannot express a *preference*, for example the fact that relation $\mathsf{p}$ is more likely than $\mathsf{pi}$. We cannot express *priorities* between the constraints either, for example the fact that the constraint $\{\mathsf{p}, \mathsf{pi}\}$ between two intervals is more important than other constraints, and that I am ready to relax some of these constraints to obtain a solution.

The construction of a fuzzy interval algebra is a response to these objectives. We will discuss Badaloni and Giacomin [2] proposal, which uses a fuzzy algebra of relations denoted by $\mathrm{IA}^{fuz}$.

**A fuzzy point algebra** The construction of the fuzzy Interval algebra is best understood if we first define a fuzzy Point algebra $\mathrm{PA}^{fuz}$:

**Definition 1.** *The domain of the algebra* $PA^{fuz}$ *is the set* $\Omega^{B_1}$ *of maps* $\alpha$: $B_1 \to [0.1]$.

The set $B_1$ contains the three relation symbols $<$, $\mathsf{eq}$, and $>$. An element $\alpha$ of the algebra may thus be identified with a 3-tuple[1] $(\alpha_1, \alpha_2, \alpha_3) = (\alpha(<), \alpha(\mathsf{eq}), \alpha(>))$ which is an element of $[0.1]^3$.

The inversion, intersection, and union operations are defined as follows:

— the inverse of $(\alpha_1, \alpha_2, \alpha_3)$ is $(\alpha_3, \alpha_2, \alpha_1)$;
— the conjunction of $\alpha$ and $\beta$ is the function $(\alpha \wedge \beta)(r) = \inf(\alpha(r), \beta(r))$;
— the disjunction of $\alpha$ and $\beta$ is the function $(\alpha \vee \beta)(r) = \sup(\alpha(r), \beta(r))$;
— the composition of $\alpha$ and $\beta$ is defined in the following way: let $\alpha$ and $\beta$ be two elements of the algebra. For each basic relation $r$, let $S_r$ be the set of pairs $(s, t)$ of basic relations such that $r \in (s \circ t)$. Then, $(\alpha \circ \beta)$ is the relation such that:

$$(\alpha \circ \beta)(r) = \sup_{(s,t) \in S_r} \inf(\alpha(s), \beta(t)).$$

**The fuzzy Interval algebra** Now Badaloni and Giacomin's definition of a fuzzy version of Allen's algebra follows the same pattern:

The domain of Allen's Interval algebra is the set $\Omega^{B_2}$, where $B_2$ is the set of the 13 symbols of Allen's relations, and $\Omega = \{0, 1\}$ is the set of truth-values, 0 (false) and 1 (true).

As in the case of the fuzzy Point algebra, we consider the fuzzy subsets of $B_2$, i.e. $\Omega$ is replaced by the set [0,1] of fuzzy "truth-values":

**Definition 2.** *The domain of the algebra* $IA^{fuz}$ *is the set* $\Omega^{B_2}$ *of maps* $\alpha$: $B_2 \to [0, 1]$.

An element of this algebra is denoted as $(\mathsf{p}[\alpha(\mathsf{p})], \ldots, \mathsf{pi}[\alpha(\mathsf{pi})])$ in [2]. If $\alpha(r) = 0$, we leave out $r[0]$. If $\alpha(r) = 1$, we denote $r[1]$ simply by $r$.

Inversion and composition are defined as follows:

The inverse $\alpha^{-1}$ of an element $\alpha$ is defined by $\alpha^{-1}(r) = \alpha(r^{-1})$. In the classical case, where $\alpha$ only takes values 0 or 1, this definition coincides with the definition of inversion.

The intersection of two elements $\alpha$ and $\beta$ is defined by $(\alpha \cap \beta)(r) = \inf(\alpha(r), \beta(r))$. Clearly, this definition reduces to the classical definition of intersection if $\alpha$ and $\beta$ are classical elements.

[1] In the literature on the subject, the usual notation for this 3-tuple is $(< [\alpha_1],\ \mathsf{eq}\ [\alpha_2], > [\alpha_3])$.

Disjunction is defined in a similar way: $(\alpha \cup \beta)(r) = \sup(\alpha(r), \beta(r))$; here again, this definition reduces the classical definition of disjunction if $\alpha$ and $\beta$ are classical elements.

Lastly, the composition of $\alpha$ and $\beta$ is defined in a way similar to that used in the case of the Point algebra: let $\alpha$ and $\beta$ be two elements of the algebra. For each Allen basic relation $r$, let $S_r$ be the set of pairs $(s, t)$ of basic relations such that $r \in (s \circ t)$. Then, $(\alpha \circ \beta)$ is the relation such that:

$$(\alpha \circ \beta)(r) = \sup_{(s,t)\in S_r} \inf(\alpha(s), \beta(t)).$$

**Constraint networks and tractability** For fuzzy constraint networks, the definition of satisfaction is modified in the following way: a degree of satisfaction is assigned to each atomic sub-network which is a consistent scenario.

**Definition 3.** *Let $\mathcal{N} = (N, C)$ be a fuzzy constraint network on the algebra $IA^{fuz}$. The **degree of satisfaction** of an atomic sub-network s: $N \times N \rightarrow B_2$ of $\mathcal{N}$ is 0 if this scenario is not consistent. If it is consistent, it is equal to $inf_{(i,j)\in N^2}\alpha(s(i, j))$.*

The set of solutions of $N$, along with the map which associates to each solution its degree of satisfaction, defines a fuzzy subset of the set of solutions whose support is contained in the set of consistent scenarios. In the particular case where the network is a classical network, the classical set of solutions is precisely the set of consistent scenarios.

**Definition 4.** *Let $\mathcal{N} = (N, C)$ be a fuzzy constraint network on the algebra $IA^{fuz}$ which is consistent. An **optimal solution** is a solution whose degree of satisfaction is equal to that of the network.*

Convex and pre-convex relations can be defined in the fuzzy case. The main result about tractability is the following:

**Theorem 7.** *The subclass of fuzzy pre-convex relations $\mathcal{H}^{fuz}$ is the unique maximal tractable subclass of $IA^{fuz}$ which contains all fuzzy basic relations.*

### 5.4. Fuzzy RCC relations

The "fuzzification" of Allen's calculus we have just described is not concerned with the temporal entities, which are still perfectly "classical" intervals. Now fuzzy entities arise naturally in many circumstances. Schockaert *et al.* illustrate their introduction of fuzzy temporal intervals using the history of painting: for example, the "Blue period" of Picasso's work is only definable in a vague manner. In the spatial domain, the necessity of being able to consider vague regions is still more obvious: what do we precisely understand by "London", or " the Los Angeles region"?

#### 5.4.1. Fuzzy regions and relations

The starting point of [55] is a universe $U$ of fuzzy regions, whose precise nature remains to be defined. A (fuzzy) region $u \in U$ is hence a function $u\colon D \rightarrow [0, 1]$, where $D$ is some domain of classical "regions".

On this universe, we consider a fuzzy binary connection relation $\mathsf{C}$: $U \times U \rightarrow [0.1]$. This fuzzy relation is assumed to be reflexive and symmetrical:

$$(\forall u \in U)\mathsf{C}(u,u) = 1 \text{ and } (\forall u, v \in U)\mathsf{C}(u,v) = \mathsf{C}(v,u).$$

The fuzzy relations deduced from C are the 11 relations defined in Table 7.2, which also recalls the classical definitions of these relations. As described above (section 7.3), the T-norm used is the Łukasiewicz T-norm, and $I_T$ indicates the implicator associated with this T-norm. It is shown in [54] that the definitions of these relations coincide with the classical definitions when the regions are classical regions, i.e. the elements of $U$ take their values from the set $\{0, 1\}$.

| **Relation** | **Classical definition** | **Fuzzy definition** |
|---|---|---|
| $\mathsf{DC}(u,v)$ | $\neg\mathsf{C}(u,v)$ | $1 - \mathsf{C}(u,v)$ |
| $\mathsf{P}(u,v)$ | $(\forall w \in U)(\mathsf{C}(w,u) \Rightarrow \mathsf{C}(w,v))$ | $\inf_{w\in U} I_T(\mathsf{C}(w,u), \mathsf{C}(w,v))$ |
| $\mathsf{PP}(u,v)$ | $\mathsf{P}(u,v) \wedge \neg\mathsf{P}(v,u)$ | $\inf(\mathsf{P}(u,v), 1 - \mathsf{P}(v,u))$ |
| $\mathsf{EQ}(u,v)$ | $\mathsf{P}(u,v) \wedge \mathsf{P}(v,u)$ | $\inf(\mathsf{P}(u,v), \mathsf{P}(v,u))$ |
| $\mathsf{O}(u,v)$ | $(\exists w \in U)\mathsf{P}(w,u) \wedge \mathsf{P}(w,v)$ | $\sup_{w\in U} T(\mathsf{P}(w,u), \mathsf{P}(w,v))$ |
| $\mathsf{DR}(u,v)$ | $\neg\mathsf{O}(u,v)$ | $1 - \mathsf{O}(u,v)$ |
| $\mathsf{PO}(u,v)$ | $\mathsf{O}(u,v) \wedge \neg\mathsf{P}(u,v) \wedge \neg\mathsf{P}(v,u)$ | $\inf(\mathsf{O}(u,v), 1 - \mathsf{P}(u,v), 1 - \mathsf{P}(v,u))$ |
| $\mathsf{EC}(u,v)$ | $\mathsf{C}(u,v) \wedge \neg\mathsf{O}(u,v)$ | $\inf(\mathsf{C}(u,v), 1 - \mathsf{O}(u,v))$ |
| $\mathsf{NTP}(u,v)$ | $(\forall w \in U)(\mathsf{C}(w,u) \Rightarrow \mathsf{O}(w,v))$ | $\inf_{w\in U} I_T(\mathsf{C}(w,u), \mathsf{O}(w,v))$ |
| $\mathsf{TPP}(u,v)$ | $\mathsf{PP}(u,v) \wedge \neg\mathsf{NTP}(u,v)$ | $\inf(\mathsf{PP}(u,v), 1 - \mathsf{NTP}(u,v))$ |
| $\mathsf{NTPP}(u,v)$ | $\mathsf{PP}(u,v) \wedge \mathsf{NTP}(u,v)$ | $\inf(1 - \mathsf{P}(v,u), \mathsf{NTP}(v,u))$ |

Table 3. Classical and fuzzy definitions of the basic RCC relations

We consider fuzzy formulas which express upper or lower bounds for the degrees of satisfaction of $R(x,y)$, where $R$ is one of the relations considered in Table 3:

**Definition 5.** *An* atomic fuzzy RCC formula *is an expression of the form* $R(x,y) \leqslant \lambda$ *or* $R(x,y) \geqslant \lambda$*, where* $R$ *is one of the 12 relations* $\mathsf{C}$, $\mathsf{DC}$, $\mathsf{P}$, $\mathsf{P}^{-1}$, *co* $\mathsf{P}$, *co*$\mathsf{P}^{-1}$, $\mathsf{O}$, $\mathsf{DR}$, $\mathsf{NTP}$, $\mathsf{NTP}^{-1}$, *co*$\mathsf{NTP}$, *and co*$\mathsf{NTP}^{-1}$.

**Definition 6.** *A* fuzzy RCC formula *is a disjunction* $\varphi \vee \ldots \vee \varphi_k$ *of atomic fuzzy* RCC *formulas.*

A technical notion of *standard* formula can be introduced.

**Definition 7.** *An* interpretation *is a map* $m$ *which associates a fuzzy region to each variable* $x \in X$*, and to* $\mathsf{C}$ *a fuzzy binary relation between regions.*

The definition of the satisfaction of a fuzzy RCC formula is what may be expected:

**Definition 8.** *An interpretation* $m$ satisfies *the atomic formula* $R(x,y) \leqslant \lambda$ *(*$R(x,y) \geqslant \lambda$*, respectively) if* $m(R)(m(x), m(y)) \leqslant \lambda$ *(*$m(R)(m(x), m(y)) \geqslant \lambda$*, respectively). It satisfies the disjunctive formula* $\varphi_1 \vee \ldots \vee \varphi_k$ *if it satisfies at least one of the* $\varphi_i$*, for* $1 \leqslant i \leqslant k$*. It satisfies a set* $\Theta$ *of formulas if it satisfies each formula in* $\Theta$*.*

**Proposition 3.** *The satisfaction problem of a set of fuzzy* RCC *formulas is NP-complete ([55], proposition 3).*

## 6. Implementing annotation schemes

### 6.1. Annotating

Annotating texts for their temporal content — or spatial content— has two sides to it: annotations can be made by humans, or it can be automated.

Automated annotation of texts, in turn, can be rule-based, where rules are based on human intuition, on linguistic knowledge, or on knowledge gained from mining big corpora, for example from Web mining.

More recently, in compliance with what Mani *et al.* [33] call the **statistical revolution** in linguistics, ML (machine learning)-based approaches have been used for automated annotation.

In ML based methods, texts which have already been annotated by humans are used to train and test programs which subsequently can be used for automated annotation tasks.

ML methods have proved to be dramatically improved when constraint-based reasoning are used to expand the training sets, as explained below.

#### 6.1.1. Rule-based annotation

A general scenario for the temporal annotation of document may be described as follows:

1. The raw text of the document is preprocessed by part-of-speech tagging and chunking.
2. Modules for EVENT, TIMEX3, TLINK, SLINK, ALINK are cascaded to output the TimeML representation.

The modules for ¡EVENT¿ and ¡TIMEX3¿ provide temporal entities, which have a temporal extent (an interval or a time-point);

The marker, or **trigger** for an ¡EVENT¿ can be a verb, a nominal, or an adjective. A ¡TIMEX3¿ tag corresponds to a date or a time: it refers to a calendar or a clock.

Relations between temporal entities are provided by the ¡TLINK¿ module (temporal relations), the ¡SLINK¿ module (subordinate clauses), and the ¡ALINK¿ module (aspect).

Let us now briefly describe two instantiations of this general scheme:

**The TARSQI system** The TARSQI project [62] “aims to enhance natural language question answering systems so that temporally-based questions about the events and entities in news articles can be addressed appropriately”. The TARSQI system can be used as a stand-alone system or a an aid for human annotators.

The system architecture is represented in the diagram of Figure 9.

The input for the system has previously been part-of-speech tagged and chunked.

– The ¡TIMEX3¿ module is called the GUTime tagger; based on TempEx, a tagger which handles both absolute times (e.g. November 27, 2011) and relative times (Sunday) by testing the local context; in most cases, a reference time is provided by the document

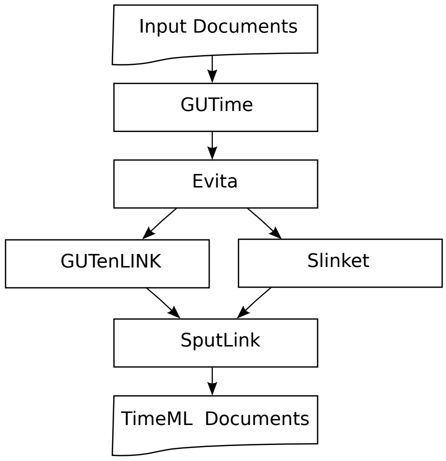


Figure 9. Architecture of the TARSQI system

publication time; it extends TempEx by handling expressions such as durations and a variety of temporal modifiers.

- The ¡EVENT¿ module is called Evita; it uses different strategies for events triggered by verbs, nouns, and adjectives. For verbs, event identification is based on a lexical look-up, aided by contextual parsing in order to exclude weak stative predicates (*be*, *have*). For nouns, a disambiguation phase accompanies the lexical look-up, which uses machine learning techniques to determine when an ambiguous noun is used with an event sense (*fair*, *demonstration*, *war* are cases in point). For adjectives, the system retains adjectives which have been pre-selected in the TimeBank corpus. Linguistic rules are applied to obtain the temporally relevant features of each element identified as an event.
- The ¡TLINK¿ module is called GUTenLINK. It uses hand-developed syntactic and lexical rules. When an event is anchored with a signal or tense/aspect cue to the event in the main clause of the previous sentence, it uses a FNT (finite state transducer) to infer the likely temporal relation. It also uses default rules for ordering events.
- The ¡SLINK¿ module is called the Slinket module. It relies on a combination of lexical and syntactic knowledge. Lexical knowledge is used to pre-select events that may introduce SLINKs. A syntactic module which is a cascade of shallow syntactic tasks is implemented as sequences of FSTs. It aims at identifying the subordinate event.

The TARSQI system also comprises a module that “takes the known temporal relations in a text and derives new implied relations from them”. This is called **temporal closure**. Basically, it uses Allen’s algebra restricted to the sub-class of pointizable relations to ensure that consistency testing is tractable.

#### 6.1.2. Annotating French texts

In [4], Bittar describes work on two modules for the recognition and annotation of temporal expressions and events in French.

The ¡TIMEX3¿ module uses a set of finite state transducers developed with the Unitex corpus processor. It also contains a transducer for detecting expressions, such as phone numbers, which are not to be considered, and are tagged as ¡GARBAGE¿. A normalization script converts the content of temporal expressions into a TimeML normalized value.

The ¡EVENT¿ tagger inputs text which has previously undergone POS tagging, an inflectional morphological analysis, and shallow syntactic analysis by the Macaon processing pipeline for French.

The ¡EVENT¿ module consists of a layer of lexical processing based on lexical look-up for noun and verb classes, and a layer of contextual processing for detecting and classifying event candidates.

The paper discusses the pros and cons of their approach — a lexical approach for the ¡TIMEX3¿ module and shallow syntactic analysis for the ¡EVENT¿ module, as compared to previous work by Parent, Gagnon, and Muller [44], who use part-of-speech tagging, morphological analysis and shallow parsing for the ¡TIMEX3¿ module, and a full dependency parse for the ¡EVENT¿ module.

#### 6.1.3. Machine Learning methods

**Pairwise ordering** Mani *et al.* [33] use ML methods to solve the following problem: given a pair (X,Y) of events or timexs in a TimeML annotated text that has been related via a TLINK by a human annotator, assign a RelType to it.

In their paper, the authors use the annotated corpus OTC which is a union of the TimeBank corpus and the Opinion Corpus. Using directly the human-annotated corpus for training a system to solve this problem behaves poorly. One reason for this behavior is the fact that humans tend to leave many temporal relations implicit, relying on the reader to fill-in the gaps when necessary. Consequently, Mani *et al.* use the algebraic apparatus of constraint-based reasoning to generate new relations which are added to the training set. This results in a dramatic increase in the number of links (the average number of tlink per event shoots from less than one to more than nine). As a consequence, the accuracy of the classification is much higher.

Detailed comparison with the use of hand-coded rules and with Google-induced lexical rules show that the method of closure of human-annotated corpora has still an advantage on both. Human hand-coded general rules may prove that some intuition-bases rules are inaccurate in some configurations of features, while many lexical rules are too specific to find some use in a specific text.

**Toward global consistency** More recently, Denis and Muller [16] discuss Mani *et al.*'s approach. According to them, it has two limitations:

- the pairs of entities to be related by the system have been pre-selected;
- each of these pair is labeled independently by a locally-trained classifier.

In Mani *et al.*'s work [33], evaluating the annotation tasks is based on checking the accuracy of prediction for the labels of chosen pairs of events: global consistency is not considered. In general, this will result in networks which are globally inconsistent.

Later work by Tatu and Srikanth [58], Bramsen *et al.* [6], Chambers and Jurafsky [8], do encode global consistency constraints. Tatu and Srikanth use a greedy search procedure with backtracking. Bramsen *et al.* propose three strategies for ordering what they call **temporal fragments**, which are “fragments of texts that do not exhibit abrupt changes in temporal focus”. Two strategies use greedy inference approaches. The third one use a reformulation of the problem as finding the optimal solution within the Linear Programming framework (ILP). Chambers and Jurafsky also use the same framework.

However Bramsen *et al.*, as well as Chambers and Jurafsky, only use a small subset of relations, namely *before* and *after*.

Denis and Muller’s approach consists in using the fact that, among all Allen relations, the subclass of convex relations can be translated in terms of (convex) relations between the endpoints of the intervals, that is, in terms of only five relations other than the empty and universal relation.

This allows these authors to reformulate the encoding of global constraints on events into integral linear programming (ILP). Off-the-shelf tools are used to solve the problem. The main advantage of the translation is that it reduces the number of relations from 82 to 5.

The other contribution of the idea (already present in Allen’s 1983 paper) of decomposing the full network into “meaningful” sub-networks and solving the consistency problem on these smaller networks. Two ways of getting such decompositions are presented: the first one exploits the fact that human annotations are often scattered, so that events appear in separate, self-connected components. The second one groups dated events appearing in the same sentence, and attaches events without dates to the most recently introduced date.

## 7. Software tools

### 7.1. The Natural Language Toolkit

The Natural Language Toolkit (NLTK) is an open source library based on the Python programming language. If includes software, data, and documentation.The following lists the main modules and functionalities of the package:

- tokenizers, sentence tokenizers, stemmers;
- collocation discovery;
- POS tagging;
- classifiers: decision treen maximum entropy, naive Bayes, EM, k-means;
- chunking: n-gram, named entity recocognition;
- parsers: chart-parsers, feature-based, unification, probabilistic, dependency;
- semantic interpretation;
- evaluation metrics: precision, recall, agreement coefficients;
- probability and estimation;
- graphical concordance, parsers, wordnet browsers, chatbots.

### 7.2. CAVaT: a tool for analyzing temporally annotated corpora

CAVaT [17] is a tool which performs corpus analysis and validation for TimeML. It is constructed from a set of Python modules and uses NLTK and MySQL. “It adds a level of abstraction between TimeML markup and human researchers, making data easier to analyze, and patterns easier to spot. It also helps identify trouble spots in annotations.”

### 7.3. Java-based tools and architectures

**Text Analytics** is the applicative side of which Natural Language Processing is the theoretical side, namely, text analytics applications are applications which extract some useful information from texts.

Many Java-based tools are available freely for download.

GATE (General Architecture for Text Engineering (`http://gate.ac.uk`)[13, 12] is a Java suite of tools for natural language processing tools. It offers **language resources**, **processing resources**, and **visual resources**.

UIMA (Unstructured Information Management Architecture) is a component software development architecture for text analytics developed by IBM and the Apache Software Foundation.

OpenNLP is a Java machine learning toolkit for NLP. The OpenNLP tools include a sentence boundary detector, a tokenizer, a POS tagger, a phrase chunker, a sentence parser, a name finder, and a co-reference resolver.

Those tools can be used independently, or as plugins with other Java frameworks, including WordFreak and UIMA.

The output of the tools is in various format according to the tools it is used with: simple text if the tools are used by themselves, WordFreak XML format if used with WordFreak, XMI format when used with UIMA.

### 7.4. Software for qualitative spatial and temporal reasoning

### 7.5. The Qualitative Algebra Toolkit (QAT)

has been developed with the aim of providing software tools for working on arbitrary qualitative formalisms without having to develop the basic tools from scratch for each new formalism. This is possible because of the similar algebraic structures of many qualitative formalisms. The QAT offers the user ready-to-use tools, once the specification of the formalism has been provided to the system.

The Qualitative Algebra Toolkit [10] (QAT) software is developed at the University of Artois, France. It is a Java library designed to offer generic tools for the definition and the manipulation of qualitative algebras and constraint networks on these algebras.

The QAT is composed of three packages: the algebra package, the qcn (*qualitative constraint networks*) package, and the solver package.

Besides the three main packages, the QAT also contains additional applicative packages. For instance, the Campaign package implements tools for making benchmarks, while the Merging package contains classes for merging the temporal or spatial information of several constraint networks.

### 7.5.1. The algebra package

This package allows the user to define arbitrary qualitative algebras (binary, ternary, or of arbitrary arity) using an XML file.

It contains ready-to-use files for several formalisms frequently used in the literature (Allen's interval algebra, Point algebra, Cyclic point algebra, Cyclic interval algebra, Rectangle algebra, INDU algebra, $n$-point algebra, $\mathsf{RCC}-5$ and $\mathsf{RCC}-8$ algebras, and Cardinal direction algebra).

It also contains tools for defining and manipulating relations.

### 7.5.2. The qcn package

This package is used to define and manipulate qualitative constraint networks on a qualitative algebra. Again, the definition of the network is provided by the user as an XML file. The package also supports the automated generation of random instances of constraint networks.

### 7.5.3. The solver package

This package contains tools for solving the main problems of interest when dealing with qualitative constraint networks: the consistency problem, the problem of finding one or all solutions, the problem of determining the minimal network.

All these methods are generic : they may be applied on arbitrary qualitative calculi.

Most of the usual solving methods are implemented, including the standard generate-and-test methods, search methods based on back-track and forward checking, and local propagation methods.

The user can configure these different methods by choosing among a range of heuristics.

Several versions of the algebraic closure algorithm are provided (PC-1, PC2, PCMix).

The QAT is an open platform which also allows the user to define and test new heuristics.

## 7.6. The SparQ toolbox

SparQ [64] is a software toolbox for qualitative reasoning. It has been developed in the context of a project on spatial cognition, and has been sponsored by the DFG (*Deutsche Forschungsgemeinschaft*). The aim of the project is to make the qualitative formalism techniques developed in the research community available in a single homogeneous framework.

SparQ contains two parts: the first part is devoted to the specification of formalisms, the second is a functional part, designed as an open set, which comprises three modules: a qualify module, a compute-relation module, and a constraint-reasoning module. Other planned modules are a neighborhood-reasoning module and a quantify module.

SparQ comprises a set of C libraries and a main program written in Lisp. It is a freeware released under the GNU (General Public License).

### 7.6.1. The qualify module

Given a (spatial) configuration and a qualitative formalism, the function of this module is to describe the configuration by using the basic relations of the formalism.

For example, if the formalism is Allen's calculus, and if the configuration contains three intervals $A = (0, 2), B = (1, 4)$, and $C = (3, 4)$, calling qualify with the option *first2all* results in the relation o between $A$ and $B$, and the relation p between $A$ and $C$. Using Lisp-like notations, we thus get the list $((A \circ B)(A\, \mathsf{p}\, C))$. With the option *all*, we would have got $((A \circ B)(A\, \mathsf{p}\, C)(B\, \mathsf{fi}\, C))$.

### 7.6.2. The compute-relation module

The function of this module is to compute the relations resulting from an operation of the calculus. Hence results are lists of relations, or sets of lists. The operations may be operations of the calculus (inversion and composition in the binary case, ternary or $n$-ary composition in general), closure operations (such as the computation of the smallest subclass generated by a set of relations), and set-theoretic operations (computing complements, intersections, unions, or differences).

### 7.6.3. A SparQ session

A typical interaction with SparQ uses a syntax in which the user specifies a functional module (for example, compute-relation, whose function is to compute a relation), a formalism (for example *allen*), and an operation (for example, compositionappropriate parameters (in the case of composition, the name of two relations of the calculus).

In this case, the user will input:

(compute-relation allen composition o d)

(the syntax used is the lisp syntax); the output of the system will be the list of relations (o s d).

### 7.6.4. The constraint-reasoning module

Given a constraint network, this module allows the user to carry out a particular operation on it. This operation may be the computation of its algebraic closure (the algorithm used is the AC-3 algorithm of [31]), or the search for an atomic refinement (a scenario) of the network which is an algebraically closed scenario (this operation is called *scenario-consistency*).

SparQ contains a module (under development) for testing the consistency of a network for formalisms whose domain of interpretation is defined in terms of real algebraic objects (this means that these objects are defined by algebraic equations on the field of real numbers). This test for consistency is based on the use of Gröbner bases [11]. This functionality can be used to build composition tables, by proving the inconsistency of unfeasible scenarios.

### 7.6.5. Available formalisms

The main formalisms available are Allen's calculus, the Cardinal relation calculus, the dependency calculus [47, 48], the Dipole calculi [53, 40, 39], the Single cross calculus [19] and the Double cross calculus [20], the Flip-flop calculus and its refinement $\mathcal{LR}$ [26, 56], the geometrical alignment calculus [18], the $\mathcal{OPRA}_m$ calculi [39, 38], the Point calculus, several versions of the Qualitative trajectory calculus [60], and the RCC−8 and RCC−5 calculi.

### 7.7. The GQR system

GQR (an acronym for *Generic Qualitative Reasoner*) [22] is a solver for binary qualitative constraint networks developed at the University of Freiburg im Breisgau, Germany. The primary focus of the designers of this reasoning software is on *genericity* and *extensibility*, while preserving *efficiency* and *scalability*.

#### 7.7.1. General principles

Given a qualitative formalism defined in a purely syntactic way by its set **B** of basic relations, and its inversion and composition operations, as well as a constraint network on the algebra **A** of relations (subsets of **B**), the aim of the software is to efficiently determine whether this network is consistent.

GQR implements Mackworth's variant of the path consistency algorithm [31, 14]. It then uses chronological backtracking, making use of known tractable subclasses of the calculus to reduce the branching factor.

In contrast to the two projects SparQ (encoded in Java) and QAT (programmed in LISP), the GQR software is encoded in C++.

The main formalisms implemented are the Point algebra, the Cardinal direction algebra (for points), Allen's interval algebra, the RCC−5 and RCC−8 formalisms, and the $\mathcal{OPRA}_4$ calculus.

#### 7.7.2. Genericity and extensibility

Genericity means that the user should be able to use the system to define and test new formalisms at will. As a consequence, he/she is offered the possibility of defining formalisms by using textual files or XML files. The user is also allowed to define and use his/her own heuristics.

In addition, the software can be used to check the algebraic properties of a formalism, and (at least partially) to compile composition tables.

#### 7.7.3. Efficiency and scalability

The techniques used in the implementation include Hogge's method used in [24] by Ladkin and Reinefeld, the use of a queue data structure as in Van Beek and Manchak's implementation of the path consistency method [59], and the use of caching techniques for computing inversion and composition [24].

The authors of [22] describe a comparison between the performances of GQR and those of two softwares designed specifically for particular calculi, namely Nebel's software [42] for Allen's interval algebra, and Renz and Nebel's software [51] for RCC−8. Both are encoded in the C language. The conclusion of their experiments is that the behavior of GQR — a generic system — although slower than that of these specialized systems, is satisfactory for easy problems, but will have to be improved in order to deal with difficult problems efficiently.

## 8. Perspectives

### 8.1. Spatio-temporal annotation and reasoning

The representation of motion has hardly been touched upon in the field of qualitative spatial reasoning, although some formalisms have been proposed to integrate spatial reasoning with temporal reasoning. Muller [41] describes a logical framework where the entities are spatio-temporal objects in space-time. Van de Weghe Qualitative trajectory calculi [60, 61, 5] compare the positions of two objects at different moments in time.

In [46], Pustejovsky and Moszkowicz propose a framework for the interpretation of motion in language which uses an annotation language called the Spatiotemporal Markup Language (STML). The representation of spatial change of an object is embedded within a first-order modal dynamic logic called Dynamic Interval Temporal Logic (DITL), which can be used to represent two ways of representing motion in language: using **path-predicates** (*Peter crossed the river.*) and using **manner-of-motion** predicates (*Peter swam across the river.*)

Devising frameworks for the integration of representation, annotation, and reasoning about spatio-temporal knowledge is is an open challenge for future research.

## References

[1] Allen, J. F. Maintaining knowledge about temporal intervals. *Comm. of the ACM 26*, 11 (1983), 832–843.

[2] Badaloni, S., and Giacomin, M. The algebra $IA^{fuz}$: a framework for qualitative fuzzy temporal reasoning. *Artif. Intell. 170*, 10 (2006), 872–908.

[3] Barkowsky, T., Knauff, M., Ligozat, G., and Montello, D. R., Eds. *Spatial Cognition V: Reasoning, Action, Interaction, International Conference Spatial Cognition 2006, Bremen, Germany, September 24-28, 2006, Revised Selected Papers* (2008), vol. 4387 of *Lecture Notes in Computer Science*, Springer.

[4] Bittar, A. Annotation of Events and Temporal Expressions in French Texts. In *Linguistic Annotation Workshop* (2009), The Association for Computational Linguistics, pp. 48–51.

[5] Bogaert, P., de Weghe, N. V., Cohn, A. G., Witlox, F., and Maeyer, P. D. The Qualitative Trajectory Calculus on Networks. In Barkowsky et al. [3], pp. 20–38.

[6] Bramsen, P., Deshpande, P., Lee, Y. K., and Barzilay, R. Inducing temporal graphs. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, 22-23 July 2006, Sydney, Australia* (2006), D. Jurafsky and É. Gaussier, Eds., ACL, pp. 189–198.

[7] Bruce, B. A model for temporal references and its application in a question answering program. *Artificial Intelligence 3*, 1–3 (1972), 1–25.

[8] Chambers, N., and Jurafsky, D. Jointly combining implicit constraints improves temporal ordering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL* (2008), ACL, pp. 698–706.

[9] Comrie, B. *Aspect: An introduction to the study of verbal aspect and related problems.* Cambridge University Press, 1976.

[10] Condotta, J.-F., Ligozat, G., and Saade, M. A generic toolkit for n-ary qualitative temporal and spatial calculi. In *The 13th International Symposium on Temporal Representation and Reasoning (TIME'06)* (Budapest, Hungary, June 2006), pp. 78–86.

[11] Cox, D. A., Little, J. B., and O'Shea, D. *Using Algebraic Geometry.* GTM. Springer, New York, 1998.

[12] Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. A framework and graphical development environment for robust NLP tools and applications. In *ACL* (2002), pp. 168–175.

[13] Cunningham, H., Wilks, Y., and Gaizauskas, R. J. GATE -a general architecture for text engineering. In *COLING* (1996), pp. 1057–1060.

[14] Dechter, R. *Constraint Processing.* Morgan Kaufmann, 2003.

[15] Dechter, R., Meiri, I., and Pearl, J. Temporal constraint networks. *Artificial Intelligence 49*, 1-3 (1991), 61–95.

[16] Denis, P., and Muller, P. Predicting globally-coherent temporal structures from texts via endpoint inference and graph decomposition. In *IJCAI* (2011), T. Walsh, Ed., IJCAI/AAAI, pp. 1788–1793.

[17] Derczynski, L., and Gaizauskas, R. J. Analysing temporally annotated corpora with cavat. In *LREC* (2010), N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, Eds., European Language Resources Association.

[18] Dylla, F., and Moratz, R. Exploiting qualitative spatial neighborhoods in the situation calculus. In Freksa et al. [21], pp. 304–322.

[19] Freksa, C. Temporal reasoning based on semi-intervals. *Artificial Intelligence 54* (1992), 129–227.

[20] Freksa, C. Using orientation information for qualitative spatial reasoning. In *Spatio-Temporal Reasoning* (1992), A. U. Frank, I. Campari, and U. Formentini, Eds., vol. 639 of *Lecture Notes in Computer Science*, Springer, pp. 162–178.

[21] Freksa, C., Knauff, M., Krieg-Brückner, B., Nebel, B., and Barkowsky, T., Eds. *Spatial Cognition IV: Reasoning, Action, Interaction, International Conference Spatial Cognition 2004, Frauenchiemsee, Germany, Revised Selected Papers* (2005), vol. 3343 of *Lecture Notes in Computer Science*, Springer.

[22] Gantner, Z., Westphal, M., and WĂ¶lfl, S. GQR- a Fast Reasoner for Binary Qualitative Constraint Calculi. In *Proc. of the AAAI'08 Workshop on Spatial and Temporal Reasoning* (Chicago, USA, 2008).

[23] Kautz, H. A., and Ladkin, P. B. Integrating metric and qualitative temporal reasoning,. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI'91)* (1991), vol. Volume 1, pp. 241–246.

[24] Ladkin, P. B., and Reinefeld, A. Fast algebraic methods for interval constraint problems. *Annals of Mathematics and Artificial Intelligence 19*, 3–4 (1997), 383–411.

[25] Li, S., and Ying, M. Region connection calculus: Its models and composition table. *Artif. Intell. 145*, 1-2 (2003), 121–146.

[26] Ligozat, G. Models for qualitative spatial reasoning. In *Proc. of the IJCAI-93 Workshop on Spatial and Temporal Reasoning* (1993), F. D. Anger, H. Güsgen, and J. van Benthem, Eds.

[27] Ligozat, G. Tractable relations in temporal reasoning: pre-convex relations. In *Proc. of the ECAI-94 Workshop on Spatial and Temporal Reasoning* (Amsterdam, 1994), F. D. Anger, H. Güsgen, and G. Ligozat, Eds., pp. 99–108.

[28] Ligozat, G. A New Proof of Tractability for ORD-Horn Relations. In *Proc. of AAAI-96* (1996), pp. 395–401.

[29] Ligozat, G., and Renz, J. What is a qualitative calculus? a general framework. In *Proc. of PRICAI'04, LNCS 3157* (Auckland, New Zealand, 2004), pp. 53–64.

[30] Ligozat, G. *Qualitative Spatial and Temporal Reasoning.* ISTE/John Wiley and Sons, 2012.

[31] Mackworth, A. K. Consistency in networks of relations. *Artificial Intelligence 8* (1977), 99–118.

[32] Mani, I., Hitzeman, J., Richer, J., Harris, D., Quimby, R., and Wellner, B. SpatialML: Annotation Scheme, Corpora, and Tools. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco* (2008), European Language Resources Association.

[33] Mani, I., Verhagen, M., Wellner, B., Lee, C. M., and Pustejovsky, J. Machine learning of temporal relations. In *Proceedings of the International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia, 17-21 July 2006* (2006), The Association for Computational Linguistics.

[34] Meiri, I. Combining qualitative and quantitative constraints in temporal reasoning. In *Proceedings of the Ninth National Conference on Artificial Intelligence* (Menlo Park, California, 1991), T. Dean and K. McKeown, Eds., American Association for Artificial Intelligence, AAAI Press, pp. 260–267.

[35] Meiri, I. Combining qualitative and quantitative constraints in temporal reasoning. *Artificial Intelligence 87*, 1–2 (1996), 343–385.

[36] Młynarczyk, A. K. *Aspectual Pairing in Polish.* PhD thesis, Universiteit Utrecht, 2004.

[37] Moens, M., and Steedman, M. Temporal Ontology and Temporal Reference. *Computational Linguistics 14*, 2 (1988), 15–28.

[38] Moratz, R. Representing relative direction as a binary relation of oriented points. In *ECAI* (2006), G. Brewka, S. Coradeschi, A. Perini, and P. Traverso, Eds., vol. 141 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, pp. 407–411.

[39] Moratz, R., Dylla, F., and Frommberger, L. A relative orientation algebra with adjustable granularity. In *Proc. of the IJCAI-05 Workshop on Agents in Real-Time and Dynamic Environments* (Edinburgh, Scotland, July 2005).

[40] Moratz, R., Renz, J., and Wolter, D. Qualitative spatial reasoning about line segments. In *ECAI* (2000), W. Horn, Ed., IOS Press, pp. 234–238.

[41] Muller, P. *ÉlĂ©ments d'une thĂ©orie du mouvement pour la formalisation du raisonnement spatio-temporel de sens commun.* PhD thesis, IRIT, UniversitĂ© Paul Sabatier, Toulouse, 1998.

[42] Nebel, B. Solving hard qualitative temporal reasoning problems: Evaluating the efficienty of using the ord-horn class. In *Proceeding of the Twelfth Conference on Artificial Intelligence (ECAI'96)* (1996).

[43] Nebel, B., and Bürckert, H.-J. Reasoning about temporal relations: A maximal tractable subclass of Allen's interval algebra. *J. of the ACM 42*, 1 (1995), 43–66.

[44] Parent, G., Gagnon, M., and Muller, P. Annotation d'expressions temporelles et d'Ă©vĂ©nements en franĂ§ais. In *Traitement Automatique des Langues Naturelles (TALN), Avignon, 09/06/08-13/06/08* (http://www.atala.org/, 2008), F. BĂ©chet, Ed., Association pour le Traitement Automatique des Langues (ATALA), p. (support Ă©lectronique).

[45] Pustejovsky, J., Ingria, R., Saurí, R., Castaño, J., Littman, J., Gaizauskas, R., Setzer, A., Katz, G., and Mani, I. The specification language TimeML. In *The Language of Time. A Reader*, I. Mani, J. Pustejovsky, and R. Gaizauskas, Eds. Oxford University Press, 2005.

[46] Pustejovsky, J., and Moszkowicz, J. L. The Qualitative Spatial Dynamics of Motion in Language. *Spatial Cognition and Computation 11*, 1 (2011), 15–44.

[47] Ragni, M., and Scivos, A. Dependency calculus: Reasoning in a general point relation algebra.

In *Proceedings of the 28th German Conference on Artificial Intelligence (KI 2005)* (Koblenz, Germany, 2005), pp. 49–63.

[48] Ragni, M., and Scivos, A. Dependency calculus reasoning in a general point relation algebra. In *IJCAI-05* (2005), L. P. Kaelbling and A. Saffiotti, Eds., Professional Book Center, pp. 1577–1578.

[49] Randell, D., Cui, Z., and Cohn, T. A spatial logic based on regions and connection. In *Proc. of KR-92* (San Mateo, CA, 1992), B. Neumann, Ed., Morgan Kaufmann, pp. 165–176.

[50] Reichenbach, H. *Elements of Symbolic Logic.* The Free Press, New York, 1966.

[51] Renz, J., and Nebel, B. Efficient Algorithms for Qualitative Spatial Reasoning. In *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI'96)* (1998), John Wiley & Sons, pp. 562–566.

[52] SaurĂ, R., Knippen, R., Verhagen, M., and Pustejovsky, J. Evita: A Robust Event Recognizer for QA Systems. In *Proceedings of HLT/EMNLP 2005* (2005), pp. 700–707.

[53] Schlieder, C. Reasoning about ordering. In *Proc. of COSIT'95* (1995), vol. 988 of *LNCS*, Springer, Berlin, Heidelberg, pp. 341–349.

[54] Schockaert, S., Cock, M. D., Cornelis, C., and Kerre, E. E. Fuzzy region connection calculus: An interpretation based on closeness. *Int. J. Approx. Reasoning 48*, 1 (2008), 332–347.

[55] Schockaert, S., Cock, M. D., and Kerre, E. E. Spatial reasoning in a fuzzy region connection calculus. *Artif. Intell. 173*, 2 (2009), 258–298.

[56] Scivos, A., and Nebel, B. The finest of its class: The natural point-based ternary calculus $\mathcal{LR}$ for qualitative spatial reasoning. In Freksa et al. [21], pp. 283–303.

[57] Stell, J. Boolean connection algebras: A New Approach to the Region-Connection Calculus. *Artificial Intelligence 122* (2000), 111–136.

[58] Tatu, M., and Srikanth, M. Experiments with reasoning for temporal relations between events. In *COLING* (2008), D. Scott and H. Uszkoreit, Eds., pp. 857–864.

[59] van Beek, P., and Manchak, D. W. The design and experimental of algorithms for temporal reasoning. *Journal of Artificial Intelligence Research 4* (1996), 1–18.

[60] Van de Weghe, N. *Representing and Reasoning about Moving Objects: A Qualitative Approach.* PhD thesis, Ghent University, 2004.

[61] Van de Weghe, N., and De Maeyer, P. Conceptual neighbourhood diagrams for representing moving objects. In *ER (Workshops)* (2005), J. Akoka, S. W. Liddle, I.-Y. Song, M. Bertolotto, I. Comyn-Wattiau, S. S.-S. Cherfi, W.-J. van den Heuvel, B. Thalheim, M. Kolp, P. Bresciani, J. Trujillo, C. Kop, and H. C. Mayr, Eds., vol. 3770 of *Lecture Notes in Computer Science*, Springer, pp. 228–238.

[62] Verhagen, M., Mani, I., Sauri, R., Littman, J., Knippen, R., Jang, S. B., Rumshisky, A., Phillips, J., and Pustejovsky, J. Automating temporal annotation with TARSQI. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005), 25-30 June 2005, University of Michigan, USA* (2005), The Association for Computational Linguistics.

[63] Vilain, M., Kautz, H. A., and van Beek, P. G. Constraint propagation algorithms for temporal reasoning: A revised report. In *Readings in Qualitative Reasoning about Physical Systems*, D. Weld and J. de Kleer, Eds. Morgan Kaufmann, 1989.

[64] Wallgrün, J. O., Frommberger, L., Wolter, D., Dylla, F., and Freksa, C. Qualitative spatial representation and reasoning in the sparq-toolbox. In Barkowsky et al. [3], pp. 39–58.

## Gérard Ligozat

Gérard Ligozat was born in France in 1944. He is a full professor (now Emeritus) of computer science at Paris-Sud University, Orsay, France and an associate researcher at CRG (Centre de Recherches en Géomatique), Université Laval, Québec, Canada. His current interest includes representing and reasoning about spatial and temporal knowledge, natural language processing, temporal and modal logics, mathematical aspects of knowledge representation, and modelling cognitive processes.

He has been educated at the École Normale Supérieure in Paris, France, and at Paris University, Ph. D. at Paris-Sud University.

He has worked at CNRS in computer science. Since 1989 he has been a professor at Paris-Sud University. He has been responsible for the research on spatial and temporal reasoning at the LIMSI laboratory. His teaching experience includes courses in mathematics, logics, computational linguistics, algorithms and data structures, databases, Artificial Intelligence and logic programming. He has been director of the graduate program "Cognitive Science" from 1991 to 2001.

His current activity aims at devising tools for integrating various types of spatial and temporal information in unified formalisms, and on the formalization of cognitive processes. His range of interest includes the study of the theoretical and computational properties of the formalisms as well as their use in application domains such as natural language processing and geographic information systems.

His publications include work in mathematics, computational linguistics and AI. He has published two books and over eighty scientific articles in international conferences and journals, and has supervised fourteen doctoral theses since 1989.

He also has had an advising activity at the international level: in the United States, in Canada, and in Poland.

He has co-organized a series of workshops in the major Artificial Intelligence conferences ( IJCAI-97, AAAI-97, ECAI-98, AAAI-98, IJCAI-99, AAAI-2000, ECAI-2000, IJCAI-2001). He also has been a member of the the European SPACENET network and has participated in the Canadian GEOIDE network (Geomatics for Informed Decisions).

His collaborations include joint work with researchers from the United States (D. Mitra, F. Anger, R. Rodriguez, R. Morris), Canada (Edwards, G., B. Moulin), Germany (B. Nebel, Renz, J.), Poland (Z. Vetulani, J. Martinek).

He has been a reviewer for several international conferences (including the IJCAI and ECAI Conferences, the TIME-97, TIME-98, TIME-99 workshops, the IEA/AIE conference, and the KR conference), as well as for several international journals (the AI Journal, Applied Intelligence, the Journal for AI Research, Spatial Cognition and Computation, and Revue Internationale de Géomatique).

He is a member of the Société Mathématique de France, the American Mathematical Society, the American Association for Artificial Intelligence, AFCET and Association Française pour l' Intelligence Artificielle (AFIA).

Published books :

H. Bestougeff et Gérard Ligozat, Outils logiques pour le traitement du temps: de la

linguistique a l'intelligence artificielle, (278p.), Masson, 1989, collection: "Études et recherches en informatique"

H. Bestougeff and Gérard Ligozat, Logical Tools for Temporal Knowledge Representation, Ellis Horwood Series in Artificial Intelligence, 1992 (an English version of the previous book).

Gérard Ligozat, Représentation des connaissances linguistiques, Armand Colin, 1994.

Gérard Ligozat, Raisonnement qualitatif sur le temps et l'espace, Hermes/Lavoisier, Paris, décembre 2010 (560 pages)

(www.decitre.fr/livres/Raisonnement-qualitatif-sur-le-temps-et-l-espace.aspx/9782746231177).