Chapter 3

Time Granularity

Jérôme Euzenat & Angelo Montanari

A temporal situation can be described at different levels of abstraction depending on the accuracy required or the available knowledge. Time granularity can be defined as the resolution power of the temporal qualification of a statement. Providing a formalism with the concept of time granularity makes it possible to model time information with respect to differently grained temporal domains. This does not merely mean that one can use different time units, e.g., months and days, to represent time quantities in a unique flat temporal model, but it involves more difficult semantic issues related to the problem of assigning a proper meaning to the association of statements with the different temporal domains of a layered temporal model and of switching from one domain to a coarser/finer one. Such an ability of providing and relating temporal representations at different "grain levels" of the same reality is both an active research theme and a major requirement for many applications (e.g., integration of layered specifications and agent communication).

After a presentation of the general requirements of a multi-granular temporal formalism, we discuss the various issues and approaches to time granularity proposed in the literature. We focus our attention on the main existing formalisms for representing and reasoning about quantitative and qualitative time granularity: the set-theoretic framework developed by Bettini et al. [Bettini *et al.*, 2000] and the logical approach systematically investigated by Montanari et al. [Montanari, 1996; Franceschet, 2002] for quantitative time granularity, and Euzenat's relational algebra granularity conversion operators [Euzenat, 2001] for qualitative time granularity. We present in detail the achieved results, we outline the open issues, and we point out the links that connect the different approaches. In the last part of the chapter, we describe some applications exploiting time granularity, and we briefly discuss related work in the areas of formal methods, temporal databases, and data mining.

3.1 Introduction

The usefulness of the addition of a notion of time granularity to representation languages is widely recognized. As an example, let us consider the problem of providing a logical specification of a wide-ranging class of real-time reactive systems whose components have dynamic behaviors regulated by very different — even by orders of magnitude — time constants (*granular systems* for short) [Montanari, 1996]. This is the case, for instance, of a pondage power station that consists of a reservoir, with filling and emptying times of days or weeks, generator units, possibly changing state in a few seconds, and electronic control devices, evolving in milliseconds or even less [Corsetti *et al.*, 1991a]. A complete specification of the power station must include the description of these components and of their interactions. A natural description of the temporal evolution of the reservoir state will probably use days: "During rainy weeks, the level of the reservoir increases 1 meter a day". The description of the control devices behavior may use microseconds: "When an alarm comes from the level sensors, send an acknowledge signal in 50 microseconds". We say that systems of such a type have *different time granularities*. It is not only somewhat unnatural, but also sometimes impossible, to compel the specifier of these systems to use a unique time granularity, microseconds in the previous example, to describe the behavior of all the components. For instance, the requirement that "the filling of the reservoir must be completed within m days" can be hardly assumed to be equivalent to the requirement that "the filling of the reservoir must be completed within n microseconds", for a suitable n (we shall discuss in detail the problems involved in such a rewriting in the next section). Since a good language must allow the specifier to easily and precisely describe all system requirements, different time granularities must be a feature of a specification language for granular systems.

A complementary point of view on time granularity is also possible: besides an important feature of a representation language, time granularity can be viewed as a formal tool to investigate the definability of meaningful timing properties, such as density and exponential grow/decay, as well as the expressiveness and decidability of temporal theories [Montanari *et al.*, 1999]. In this respect, the number and organization of layers (single vs. multiple, finite vs. infinite, upward unbounded vs. downward unbounded) of the underlying temporal structure plays a major role: certain timing properties can be expressed using a single layer; others using a finite number of layers; others only exploiting an infinite number of layers. In particular, finitely-layered metric temporal logics can be used to specify timing properties of granular systems composed by a finite number of differently-grained temporal components, which have been fixed once and for all (n-layered temporal structures). Furthermore, if provided with a rich enough layered structure, they suffice to deal with conditions like "p holds at all even times of a given temporal domain" that cannot be expressed using flat propositional temporal logics [Emerson, 1990] (as a matter of fact, a 2-layered structure suffices to capture the above condition). ω-layered metric temporal logics allow one to express relevant properties of infinite sequences of states over a single temporal domain that cannot be captured by using flat or finitely-layered temporal logics. This is the case, for instance, of conditions like "p holds at all times 2^i, for all natural numbers i, of a given temporal domain".

The chapter is organized as follows. In Section 3.2, we introduce the general requirements of a multi-granular temporal formalism, and then we discuss the different issues and approaches to time granularity proposed in the literature. In Sections 3.3 and 3.4, we illustrate in detail the two main existing formal systems for representing and reasoning about quantitative time granularity: the set-theoretic framework for time granularity developed by Bettini et al. [Bettini *et al.*, 2000] and the logical approach systematically explored by Montanari et al. [Montanari, 1996; Franceschet, 2002]. In Section 3.5, we present the relational algebra granularity conversion operators proposed by [Euzenat, 2001] to deal with qualitative time granularity and we briefly describe the approximation framework outlined by Bittner [Bittner, 2002]. In Section 3.6, we describe some applications exploiting time granularity, while in Section 3.7 we briefly discuss related work. The concluding remarks provide an assessment of the work done in the field of time granularity and give an indication of possible research directions.

3.2 General setting for time granularity

In order to give a formal meaning to the use of different time granularities in a representation language, two main problems have to be solved: the qualification of statements with respect to time granularity and the definition of the links between statements associated with a given time granularity, e.g., *days*, and statements associated with another granularity, e.g., *microseconds* [Montanari, 1996]. Sometimes, these problems have an obvious solution that consists in using *different time units* — say, months and minutes — to measure time quantities in a *unique model*. In most cases, however, the treatment of different time granularities involves more difficult semantic problems. Let consider, for instance, the sentence: "every month, if an employee works, then he gets his salary". It could be formalized, in a first-order language, by the following formula:

$$\forall t_m, emp(work(emp, t_m) \rightarrow get\_salary(emp, t_m)),$$

with an obvious meaning of the used symbols, once it is stated that the subscript m denotes the fact that t is measured by the time unit of *months*.
Another requirement can be expressed by the sentence: "an employee must complete every received job within 3 days". It can be formalized by the formula:

$$\forall t_d, emp, job(get\_job(emp, job, t_d) \rightarrow job\_done(emp, job, t_d + 3)),$$

where the subscript d denotes that t is measured by the time unit of *days*.
Assume now that the two formulas are part of the specification of the same office system. We need a *common model* for both formulas. As done before, we could choose the finest temporal domain, i.e., the set of (times measured by) *days*, as the common domain. Then, a term labeled by m would be translated into a term labeled d by multiplying its value by 30. However, the statement "every month, if an employee works, then he gets his salary" is clearly different from the statement "every day, if an employee works, then he gets his salary". In fact, working for a month means that one works for 22 days in the month, whereas getting a monthly salary means that there is one day when one gets the salary for the month. Similarly, stating that "every day of a given month it rains" does not mean, in general, that "it rains for all seconds of all days of the month". On the contrary, if one states that "a car has been moving for three hours at a speed greater than 30 km per hour", he usually means that for all seconds included in the considered three hours the car has been moving at the specified speed. The above examples show that the interpretations of temporal statements are likely to change when switching from one time granularity to another one. The addition of the concept of time granularity is thus necessary to allow one to build granular temporal models by referring to the natural scale in any component of the model and by properly constraining the interactions between differently-grained components.

Further difficulties arise from the *synchronization problem* of temporal domains [Corsetti *et al.*, 1991a]. Such a problem can be illustrated by the following examples. Consider the sentence "tomorrow I will eat". If one interprets it in the domain of hours, its meaning is that there will be several hours, starting from the next midnight until the following one, when it will be true that I eat, *no matter in which hour of the present day this sentence is claimed*.

Thus, if the sentence is claimed at 1 a.m., it will be true that "I eat" at some hours t whose distance d from the current hour is such that $23 \leq d < 47$. Instead, if the same sentence is claimed at 10 p.m. of the same day, d will be such that $2 \leq d < 26$. Consider now the sentence "dinner will be ready in one hour". If it is interpreted in the domain of minutes, its meaning is that dinner will be ready in 60 minutes *starting from the minute when it is claimed*. Therefore, if the sentence is claimed at minute, say, 10, or 55, of a given hour, it will be always true that "dinner is ready" at a minute t whose distance d from the current minute is *exactly* 60 minutes. Clearly, the two examples require two different semantics.

Thus, when the granularity concept is applied to time, we generally assume a set of differently-grained domains (or layers) with respect to which the situations are described and some operators relating the components of the multi-level description. The resulting system will depend on the language in which situations are modeled, the properties of the layers, and the operators. Although these elements are not fully independent, we first take into consideration each of them separately.

3.2.1 Languages, layers, operators

The distinctive features of a formal system for time granularity depend on some basic decisions about the way in which one models the relationships between the representations of a given situation with respect to different granularity layers.

Languages. The first choice concerns the language. One possibility is to use the same language to describe a situation with respect to different granularity layers. As an example, the representations associated with the different layers can use the same temporal logic or the same algebra of relations. In such a way, the representations of the same situation at different abstraction levels turn out to be *homogeneous*. Another possibility is to use different languages at different levels of abstraction, thus providing a set of *hybrid* representations of the same situation. As an example, one can adopt a metric representation at the finer layers and a qualitative one at the coarser ones.

Layers. Any formal system for time granularity must feature a number of different (granularity) layers. They can be either explicitly introduced by means of suitable linguistic primitives or implicitly associated with the different representations of a given situation.

Operators. Another choice concerns the operators that the formal system must encompass to deal with the layered structure. In this respect, one must make provision for at least two basic operators:

contextualization to select a layer;

projection to move across layers.

These operators are independent of the specific formalism one can adopt to represent and to reason about time granularity, that is, each formalism must somehow support such operators. They are sufficient for expressing fundamental questions one would like to ask to a granular representation:

- converting a representation from a given granularity to another one (how would a particular representation appear under a finer or coarser granularity?);
- testing the compatibility of two representations (is it possible that they represent the same situation under different granularities?);
- comparing the relative granularities of two representations (which is the coarser/finer representation of a given situation?).

Internal vs. external layers. Once the relevance of these operators is established, it must be decided if the granularity applies within a formalism or across formalisms. In other terms, it must be decided if an existing formalism will be extended with these new operators or if these operators will be defined and applied from the outside to representations using existing formalisms. Both these alternatives have been explored in the literature:

- Some solutions propose an internal extension of existing formalisms to explicitly introduce the notion of granularity layer in the representations (see Sections 3.4.1 and 3.4.2 [Ciapessoni *et al.*, 1993; Montanari, 1996; Montanari *et al.*, 1999]), thus allowing one to express complex statements combining granularity with other notions. The representations of a situation with respect to different granularity layers in the resulting formalism are clearly homogeneous.
- Other solutions propose an external apprehension which allows one to relate two descriptions expressed in the same formalism or in different formalisms (see Sections 3.3, 3.4.3, and 3.5 [Euzenat, 1995b; Fiadeiro and Maibaum, 1994; Franceschet, 2002; Franceschet and Montanari, 2004]). This solution has the advantage of preserving the usual complexity of the underlying formalism, as far as no additional complexity is introduced by granularity.

3.2.2 Properties of languages

The whole spectrum of languages for representing time presented in this book is available for expressing the sentences subject to granularity. Here we briefly point out some alternatives that can affect the management of granularity.

Qualitative and quantitative languages. There can be many structures on which a temporal representation language is grounded. These structures can be compared with that of mathematical spaces:

set-theory when the language takes into account containment (i.e. set-membership);

topology when the language accounts for connexity and convexity;

metric spaces when the language takes advantage of a metric in order to quantify the relationship (distance) between temporal entities.

vector spaces when the language considers alignment and precedence (with regard to an alignment). As far as time is considered as totally ordered, the order comes naturally.

A quantitative representation language is generally a language which embodies properties of metric and vector spaces. Such a language allows one to precisely define a displacement operator (of a particular distance along an axis). A qualitative representation language does not use a metric and thus one cannot precisely state the position of objects. For instance, Allen's Interval Algebra (see Chapter 1) considers notions from vector (before) and topological (meets) spaces.

Expressive power. The expressive power of the languages can vary a lot (this is true in general for classical temporal representation languages, see Chapter 6). It can roughly be:

exact and conjunctive when each temporal entity is localized at a particular known position (a is ten minutes after b) and a situation is described by a conjunction of such sentences;

propositional when the language allows one to express conjunction and disjunction of propositional statements (a is before or after b); this also applies to constrained positions of entities (a is between ten minutes and one hour after b);

first-order when the language contains variables which allow one to quantify over the entities (there exists time lap x in between a and b);

"second-order" when the language contains variables which allow one to quantify over layers (there exists a layer g under which a is after b).

3.2.3 Properties of layers

As it always happens when time information has to be managed by a system, the properties of the adopted model of time influence the representation. The distinctive feature of the models of time that incorporate time granularity is the coexistence of a set $\mathcal{T}$ of temporal domains. Such a set is called *temporal universe* and the temporal domains belonging to it are called (temporal) *layers*. Layers can be either overlapping, as in the case of `Days` and `Working Days`, since every working day is a day (cf. Section 3.3), or disjoint, as in the case of `Days` and `Weeks` (cf. Section 3.4).

Structure of time. It is apparent that the temporal structure of the layers influences the semantics of the operators. Different structures can obviously be used. Moreover, one can either constrain the layers to share the same structure or to allow different layers to have different structures.
For each layer $T \in \mathcal{T}$, let $<$ be a linear order over the set of time points in T. We confine our attention to the following temporal structures:

continuous T is isomorphic to the set of real numbers (this is the usual interpretation of time);

dense between every two different points there is a point

$$\forall x, y \in T\ \exists z \in T (x < y \rightarrow x < z < y);$$

discrete every point having a successor (respectively, a predecessor) has an immediate one

$$\forall x \in T((\exists y \in T(x < y) \rightarrow \exists z \in T(x < z \wedge \forall w \in T\, \neg(x < w < z))) \wedge$$
$$(\exists y \in T(y < x) \rightarrow \exists z \in T(z < x \wedge \forall w \in T\, \neg(z < w < x)))).$$

Most formal systems for time granularity assume layers to be discrete, with the possible exception of the most detailed layer, if any, whose temporal structure can be dense, or even continuous (an exception is [Endriss, 2003]). The reason of this choice is that each dense layer is already at the finest level of granularity, and it allows any degree of precision in measuring time. As a consequence, for dense layers one must distinguish granularity from metric, while, for discrete layers, one can define granularity in terms of set cardinality and assimilate it to a natural notion of metric. Mapping, say, a set of rational numbers into another set of rational numbers would only mean changing the unit of measure with no semantic effect, just in the same way one can decide to describe geometric facts by using, say, kilometers or centimeters. If kilometers are measured by rational numbers, indeed, the same level of precision as with centimeters can be achieved. On the contrary, the key point in time granularity is that saying that something holds for all days in a given interval does not imply that it holds at every second belonging to the interval [Corsetti *et al.*, 1991a]. For the sake of simplicity, in the following we assume each layer to be discrete.

Global organization of layers. Further conditions can be added to constrain the global organization of the set of layers. So far, layers have been considered as independent representation spaces. However, we are actually interested in comparing their grains, that is, we want to be able to establish whether the grain of a given layer is finer or coarser than the grain of another one. It is thus natural to define an order relation $\prec$, called *granularity* relation, on the set of layers of $\mathcal{T}$ based on their grains: we say that a layer T is finer (resp. coarser) than a layer T', denoted by $T \prec T'$ (resp. $T' \prec T$), if the grain of T is finer (resp. coarser) than that of T'. There exist at least three meaningful cases:

partial order $\prec$ is a reflexive, transitive, and anti-symmetric relation over layers;

(semi-)lattice $\prec$ is a partial order such that, given any two layers $T, T' \in \mathcal{T}$, there exists a layer $T \wedge T' \in \mathcal{T}$ such that $T \wedge T' \prec T$ and $T \wedge T' \prec T'$, and any other layer T" with the same property is such that $T" \prec T \wedge T'$;

total order $\prec$ is a partial order such that, for all $T, T' \in \mathcal{T}$, either $T = T'$ or $T \prec T'$ or $T' \prec T$.

We shall see that the set of admissible operations on layers depends on the structure of $\prec$.

Beside the order relation $\prec$, one must consider the cardinality of the set $\mathcal{T}$. Even though a finite number of layers suffices for many applications, there exist significant properties that can be expressed only using an infinite number of layers (cf. Section 3.4.2). As an example, an infinite number of arbitrarily fine (discrete) layers makes it possible to express properties related to temporal density, e.g., the fact that two states are distinct, but arbitrarily close.

Pairwise organization of layers. Even in the case in which layers are totally ordered, their organization can be made more precise. For instance, consider the case of a situation described with respect to the totally ordered set of granularities including `years`, `months`, `weeks`, and `days`. The relationships between these layers differ a lot. Such differences can be described through the following notions:

homogeneity when the (temporal) entities of the coarser layer consist of the same number of entities of the finer one;

alignment when the entities of the finer layer are mapped in only one entity of the coarser one.

These two notions allow us to distinguish four different cases:

year-month the situation is very neat between years and months since each year contains the same number of months (homogeneity) and each month is mapped onto only one year (alignment);

year-week a year contains a various number of weeks (non homogeneity) and a week can be mapped into more than one year (non alignment);

month-day while every day is mapped into exactly one month (alignment), the number of days in a month is variable (non homogeneity);

working week-day one can easily imagine working weeks beginning at 5 o'clock on Mondays (this kind of weeks exists in industrial plants): while every week is made of the same duration or amount of days (homogeneity), some days are mapped into two weeks (non alignment).

How the objects behave. There are several options with regard to the behavior of the objects considered by the theories. The objects can

persist when they remain the same across layers (in the logical setting, this is modeled by the Barcan formula);

change category when, moving from one layer to another one, they are transformed into objects of different size (e.g., transforming intervals into points, or vice versa, or changing an object into another of a bigger/lower dimension, see Section 3.6.4);

vanish when an object associated with a fine layer disappears in a coarser one.

3.2.4 Properties of operators

The operator that models the change of granularity is the *projection* operator. It relates the temporal entities of a given layer to the corresponding entities of a finer/coarser layer. In some formal systems, it also models the change of the interpretation context from one layer to another. The projection operator is characterized by a number of distinctive properties, including:

reflexivity (see Section 3.5.2 self-conservation p. 105 and Section 3.4.1 p. 85) constrains an entity to be able to be converted into itself;

symmetry (see Section 3.5.2 inverse compatibility p. 106 and Section 3.4.1 p. 85) states that if an entity can be converted into another one, then this latter entity can be converted back into the original one;

order-preservation (for vectorial systems, see Section 3.3 p. 69, Section 3.5.2 p. 105, and Section 3.4.1 p. 86) constrains the projection operators to preserve the order of entities among layers;

transitivity (see below) constrains consecutive applications of projection operators in any "direction" to yield the same result as a direct projection;

oriented transitivity (see Section 3.5.2 p. 106 and Section 3.4.1 downward transitivity p. 85 and upward transitivity p. 86) constrains successive applications of projection operators in the same "direction" to yield the same result as a direct projection;

downward/upward transitivity (see Section 3.4.1 pp. 85-86 and [Euzenat, 1993]) constrains two consecutive applications of the projection operators (first downward, then upward) to yield the same result as a direct downward or upward projection;

Some properties of projection operators are related to pairwise properties of layers:

contiguity (see Section 3.4.1 p. 86), or "contiguity-preservation", constrains the projections of two contiguous entities to be either two contiguous (sets of) entities or the same entity (set of entities);

total covering (see Section 3.3 p. 69 and Section 3.4.1 p. 86) constrains each layer to be totally accessible from any other layer by projection;

convexity (see Section 3.4.1 p. 86) constrains the coarse equivalent of an entity belonging to a given layer to cover a convex set of entities of such a layer;

synchronization (see Sections 3.3 and 3.4.1), or "origin alignment", constrains the origin of a layer to be projected on the origin of the other layers. It is called synchronization because it is related to "synchronicity" which binds all the layers to the same clock;

homogeneity (see Section 3.4.1 p. 86) constrains the temporal entities of a given layer to be projected on the same number of entities of a finer layer;

Such properties are satisfied when they are satisfied by all pairs of layers.

3.2.5 Quantitative and qualitative models

In the following we present in detail the main formal systems for time granularity proposed in the literature. We found it useful to make a distinction between quantitative and qualitative models of time granularity. Quantitative models are able to position temporal entities (or occurrences) within a metric frame. They have been obtained following either a set-theoretic approach or a logical one. In contrast, qualitative models characterize the position of temporal entities with respect to each other. This characterization is often topological or vectorial. The main qualitative approach to granularity is of algebraic nature.

The set-theoretic approach is based upon naive set theory and algebra. According to it, the single temporal domain of flat temporal models is replaced by a temporal universe, which is defined as a set of inter-related temporal layers, which is built upon its finest layer. The finest layer is a totally ordered set, whose elements are the smallest temporal units relevant to the considered application (*chronons*, according to the database terminology [Dyreson and

Snodgrass, 1994; Jensen *et al.*, 1994]); any coarser layer is defined as a suitable partition of this basic layer. To operate on elements belonging to the same layer, the familiar Boolean algebra of subsets suffices. Operations between elements belonging to different layers require a preliminary mapping to a common layer. Such an approach, originally proposed by Clifford and Rao in [Clifford and Rao, 1988], has been successively refined and generalized by Bettini et al. in a number of papers [Bettini *et al.*, 2000]. In Section 3.3, we shall describe the evolution of the set-theoretic approach to time granularity from its original formulation up to its more recent developments.

According to the logical approach, the single temporal domain of (metric) temporal logic is replaced by a temporal universe consisting of a possibly *infinite* set of inter-related differently-grained layers and logical tools are provided to qualify temporal statements with respect to the temporal universe and to switch temporal statements across layers. Logics for time granularities have been given both non-classical and classical formalizations. In the non-classical setting, they have been obtained by extending metric temporal logics with operators for temporal contextualization and projection [Ciapessoni *et al.*, 1993; Montanari, 1996; Montanari and de Rijke, 1997], as well as by combining linear and branching temporal logics in a suitable way [Franceschet, 2002; Franceschet and Montanari, 2003; Franceschet and Montanari, 2004]. In the classical one, they have been characterized in terms of (extensions of) the well-known monadic second-order theories of k successors and of their fragments [Montanari and Policriti, 1996; Montanari *et al.*, 1999; Franceschet *et al.*, 2003]. In Section 3.4, we shall present in detail both approaches.

The study of granularity in a qualitative context is presented in Section 3.5. It amounts to characterize the variation of relations between temporal entities that are induced by granularity changes. A number of axioms for characterizing granularity conversion operators have been provided in [Euzenat, 1993; Euzenat, 1995a], which have been later shown to be consistent and independent [Euzenat, 2001]. Granularity operators for the usual algebras of temporal relations have been derived from these axioms. Another approach to characterizing granularity in qualitative relations, associated with a new way of generating systems of relations, has recently come to light [Bittner, 2002]. The relations between two entities are characterized by the relation (in a simpler relation set) between the intersection of the two entities and each of them. Temporal locations of entities are then approximated by subsets of a partition of the temporal domain, so that the relation between the two entities can itself be approximated by the relation holding between their approximated locations. This relation (that corresponds to the original relation under the coarser granularity) is obtained directly by maximizing and minimizing the set of possible relations.

3.3 The set-theoretic approach

In this section, we present several contributions to the development of a general framework for time granularity coming from both the area of knowledge-based systems and that of database systems. We qualify their common approach as set-theoretic because it relies on a temporal domain defined as an ordered set, it builds granularities by grouping subsets of this domain, and it expresses their properties through set relations and operations over sets. In the area of knowledge representation and reasoning, the addition of a notion of time granularity to knowledge-based systems has been one of the most effective attempts at dealing with the widely recognized problem of managing periodic phenomena. Two relevant set-theoretic approaches to time granularity are the formalism of collection expressions proposed by Leban et al. [Leban *et al.*, 1986] and the formalism of slice expressions developed by Niezette and Stevenne [Niézette and Stevenne, 1992]. In the database area, time granularity emerged as a formal tool to deal with the intrinsic characteristics of calendars in a principled way. The set-theoretic approach to time granularity was originally proposed by Clifford and Rao [Clifford and Rao, 1988] as a suitable way of structuring information with a temporal dimension, independently of any particular calendric system, and, later, it has been systematically explored by Bettini et al. in a series of papers [Bettini *et al.*, 1998a; Bettini *et al.*, 1998b; Bettini *et al.*, 1996; Bettini *et al.*, 1998c; Bettini *et al.*, 1998d; Bettini *et al.*, 2000]. As a matter of fact, the set-theoretic framework developed by Bettini et al. subsumes all the other ones. In the following, we shall briefly describe its distinctive features. A comprehensive presentation of it is given in [Bettini *et al.*, 2000]

3.3.1 Granularities

The basic ingredients of the set-theoretic approach to time granularity have been outlined in Clifford and Rao's work. Even though the point of view of the authors has been largely revised and extended by subsequent work, most of their original intuitions have been preserved.

The temporal structure they propose is a temporal universe consisting of a finite, totally ordered set of temporal domains built upon some base discrete, totally ordered, infinite set which represents the smallest observable/interesting time units.

Let T^0 be the chosen base temporal domain. A temporal universe $\mathcal{T}$ is a finite sequence $\langle T^0, T^1, \ldots, T^n \rangle$ such that, for $i, j = 0, 1, \ldots, n$, if $i \neq j$, then $T^i \cap T^j = \emptyset$, and, for $i = 0, 1, \ldots, n-1$, T^{i+1} is a constructed intervallic partition of T^i. We say that T^{i+1} is a constructed intervallic partition of T^i if there exists a mapping $\psi_i^{i+1} : T^{i+1} \rightarrow 2^{T^i}$ which satisfies the following two properties: (i) $\psi_i^{i+1}(x)$ is a (finite) convex subset of T^i (convexity) , and (ii) $\bigcup_{x \in T^{i+1}} \psi_i^{i+1}(x) = T^i$ (total covering). If we add the conditions that, for each $x \in T^{i+1}$, $\psi_i^{i+1}(x) \neq \emptyset$ and, for every pair $x, y \in T^{i+1}$, with $x \neq y$,$\psi_i^{i+1}(x) \cap \psi_i^{i+1}(y) = \emptyset$, the temporal domain T^{i+1}, under the mapping ψ_i^{i+1}, can be viewed as a partition of T^i. Furthermore, the resulting mapping ψ_i^{i+1} allows us to inherit a total order of T^{i+1} from the total order of T^i as follows (order-preservation). Given a finite closed interval S of T^i, let $first(S)$ and $last(S)$ be respectively the first and the last element of S with respect to the total order of T^i. A total order of T^{i+1} can be obtained by stating that, for all $x, y \in T^{i+1}$, $x < y$ if and only if $last(\psi_i^{i+1}(x)) < first(\psi_i^{i+1}(y))$.

In [Bettini *et al.*, 1998c; Bettini *et al.*, 1998d; Bettini *et al.*, 2000], Bettini et al. have generalized that simple temporal structure for time granularity. The framework they propose is based on a *time domain* $\langle T, \leq \rangle$, that is, a totally ordered set, which can be dense or discrete. A *granularity* g is a function from an index set I_g to the powerset of T such that:

$$\forall i, j, k \in I_g (i < k < j \wedge g(i) \neq \emptyset \wedge g(j) \neq \emptyset \Rightarrow g(k) \neq \emptyset) \qquad \text{(conservation)}$$
$$\forall i, j \in I_g (i < j \Rightarrow \forall x \in g(i)\ \forall y \in g(j)\ x < y) \qquad \text{(order preservation)}$$

Typical examples of granularities are the business weeks which map week numbers to sets of five days (from Monday to Friday) and ignore completely Saturday and Sunday. I_g can be any discrete ordered set. However, for practical reasons, and without loss of generality, we shall consider below that it is either $\mathbb{N}$ or an interval of $\mathbb{N}$.

The *origin* of a granularity is $g_0 = g(min_<(I_g))$ and its *anchor* is $a \in g_0$ such that $\forall x \in g_0(a \leq x)$. The *image* of a granularity g is $Im(g) = \cup_{i \in I_g} g(i)$ and its *extent* is $Ext(g) = \{x \in T : \exists a, b \in Im(g)(a \leq x \leq b)\}$. Two granules $g(i)$ and $g(j)$ are said to be *contiguous* if and only if $\nexists x \in T(g(i) \leq x \leq g(j))$.

3.3.2 Relations between granularities

One of the important aspects of the work by Bettini et al. is the definition of many different relationships between granularities:

$$g \trianglelefteq h \equiv \forall j \in I_h, \exists S \subseteq I_g(h(j) = \cup_{i \in S} g(i)) \quad (g \text{ groups into } h)$$

$$g \preceq h \equiv \forall i \in I_g, \exists j \in I_h(g(i) \subseteq h(j)) \quad (g \text{ is finer than } h)$$

$$g \sqsubseteq h \equiv \forall i \in I_g, \exists j \in I_h(g(i) = h(j)) \quad (g \text{ is a subgranularity of } h)$$

$$g \leftrightarrow h \equiv \exists k \in \mathbb{N}\, \forall i \in I_g(g(i) = h(i+k)) \quad (g \text{ is shift-equivalent to } h)$$

$$g \trianglelefteq h \text{ and } g \preceq h \quad (g \text{ partitions } h)$$

$$g \widehat{\subseteq} h \equiv Im(g) \subseteq Im(h) \quad (g \text{ is covered by } h)$$

$$g \trianglelefteq h \text{ and } \exists r, p \in \mathbb{Z}^+(r \leq |I_h| \wedge \forall i \in I_h(h(i) = \cup_{x=0}^{k} g(j_x) \wedge h(i+r) \neq \emptyset \Rightarrow h(i+r) = \cup_{x=0}^{k} g(j_x + p))) \quad (g \text{ groups periodically into } h)$$

Apart from the case of shift-equivalence, all these definitions state, in different ways, that g is a more precise granularity than h. As an example, the groups into relation groups together intervals of g. In fact, it can groups a subset of the elements within the interval, but in such a case the excluded elements cannot belong to any other granule of the less precise granularity. Finer than requires that all the granules of g are covered by a granule of h. So h can group granules of g, but never forget one. However, it can introduce granules that were not taken into account by g (between two g-granules). Sub-granularity can only do exactly that (i.e., it cannot group g-granules). Shift-equivalence is, in spirit, the relation holding between two granularities that are equivalent up to index renaming. It is here restricted to integer increment. Partition, as we shall see below, is the easy-behaving relationship in which the less precise granularity is just a partition of the granules of the more precise one.

It is noteworthy that all these relationships consider only aligned granularities, that is, the granules of the more precise granularity are either preserved of forgotten, but never broken, in the less precise one.

These relations are ordered by strength as below.

Proposition 3.3.1. $\forall h, g(g \sqsubseteq h \Rightarrow g \preceq h \Rightarrow g \widehat{\subseteq} h)$

It also appears that the shift-equivalence is indeed the congruence relation induced by the subgranularity relation.

Proposition 3.3.2. $\forall h, g(g \leftrightarrow h$ *iff* $g \sqsubseteq h$ *and* $h \sqsubseteq g)$

It is an equivalence relation and if we consider the quotient set of granularity modulo shift-equivalence, then $\sqsubseteq$ but also $\preceq$ and $\trianglelefteq$ define partial orders (and thus partition as well) and $\widehat{\subseteq}$ is still a pre-order.

3.3.3 Granularity systems and calendars

For the purpose of using the granularities, it is more convenient to study granularity systems, i.e., sets of granularities related by different constraints.

A *calendar* is a set S of granularities over the same time domain that includes a granularity g such that $\forall h \in S(g \trianglelefteq h)$. Considering sets of granularities in which items can be converted, there are four important design choices:

The choice of the absolute time set $\mathcal{A}$ dense, discrete or continuous.

Restriction on the use of the index set if it is common to all granularities, otherwise, the restriction hold between them; the authors offer the choice between $\mathbb{N}$ or $\mathbb{N}^+$. More generally, the choice can be done among index sets isomorphic to these.

Constraints on the granularities no gaps within a granule, no gaps between granules, no gaps on left/right (i.e., the granularity covers the whole domain), with uniform extent.

Constraints between granularities which can be expressed through the above-defined relationships.

They define, as their reference granularity frame, the *General Granularity on Reals* by:

- Absolute time is the set $\mathbb{R}$;
- index set is $\mathbb{N}^+$;
- no restrictions on granules;
- no two granularities are in shift-equivalent.

Two particular units $g_\top$ and $g_\bot$ can be defined such that:

$$\forall i \in \mathbb{N}^+, g_\bot(i) = \emptyset \text{ and } g_\top(i) = \begin{cases} T & \text{if } i = 1; \\ \emptyset & \text{otherwise.} \end{cases}$$

It is shown [Bettini *et al.*, 1996] that under sensible assumptions (namely, order-preservation or convexity-contiguity-totality), the set of units is a lattice with respect to $\preceq$ in which $g_\top$ (resp. $g_\bot$) is the greatest (resp. lowest) element. In [Bettini *et al.*, 2000], it is proved that this applies to any granularity system having no two granularity shift-equivalence (i.e., $\leftrightarrow = \emptyset$). This is important because any granularity system can be quotiented by shift-equivalence.

Finally, two conversion operators on the set of granularities are defined. The *upward conversion* between granularities is defined as:

$$\forall i \in I_g, \uparrow_g^h i = \begin{cases} j & \text{if } \exists j \in I_h(g(i) \subseteq h(j)); \\ undefined & \text{otherwise.} \end{cases}$$

Notice that the upward operator is thus only defined in the aligned case expressed by the "finer than" relationship.

Proposition 3.3.3. *if $g \preceq h$, then $\uparrow_g^h$ is always defined.*

The *downward conversion* between granularities is defined as:

$$\forall j \in I_h, \downarrow_g^h j = \begin{cases} \langle i,k \rangle & \text{if } h(j) = \cup_{x=i}^{i+k-1} g(x); \\ undefined & \text{otherwise.} \end{cases}$$

The result is thus the set of elements covered by $h(j)$. Obviously, here the "groups into" relation between the granularities ensures the totality of the downward conversion.

Proposition 3.3.4. *if $g \trianglelefteq h$, then $\downarrow_g^h$ is always defined.*

3.3.4 Algebra for generating granularities

As it is usual in the database tradition, the authors investigate the many ways in which granularities can be freely generated by applying operations to other granularities. This can be used for defining the free generated system from a set of base granularities over the same temporal domain and a set of operations. With these operations will naturally come corresponding conversion operators.

Two set of operations are identified: grouping (or group-oriented) operations, which create a granularity by grouping granules of another granularity, and selection (or granule-oriented) operations, which create a granularity by selecting granules of another granularity.

These operations are informally described below. Interested readers must refer to [Bettini *et al.*, 2000] which adds new notions (label-aligned subgranularities) for facilitating their introduction.

Grouping operations are the following:

$group_m(g)$ groups m granules of a granularity g into one granule of granularity $group_m(g)$;

$alter^m_{l,k}(g, g')$ modifies granularity g such that any l^{th} granule having k additional granules of g' (g' must partition g, k can be negative);

$shift_m(g)$ creates a granularity shift-equivalent to g modulo m;

$combine(g, h)$ creates a new granularity whose granules group granules of h belonging to the same granule of g;

$anchor - group(g, h)$ creates a new granularity by adding to each granule of h all following granules of g before the next granule of h.

Selection operations are the following:

$subset^n_m(g)$ selects the granules of g whose index are between m and n;

$select - up(g, h)$ selects the granules of g that contain at least one granule of h;

$select - down^l_k(g, h)$ selects the l granules of g starting with the k^{th} in each granule of h;

$select - by - intersect^l_k(g, h)$ selects the k granules of g starting with the l^{th} in each ordered set of granules intersecting any granule of h;

$union(g, h)$, $intersection(g, h)$, $difference(g, h)$ are defined as the corresponding operations on the set of granules of two subgranularities of the same reference granularity.

A consequence of the choice of these operations is that the operators never create finer granularities from coarser ones (they either group granules for a coarser granularity or select a subset of the granules of one existing granularity). This can be applied, for instance, generating many granularities starting with the `second` granularity (directly inspired from [Bettini *et al.*, 2000]):

$$
\begin{aligned}
\texttt{minute} &= group_{60}(\texttt{second})\\
\texttt{hour} &= group_{60}(\texttt{minute})\\
\texttt{USEasthour} &= shift_{-5}(\texttt{hour})\\
\texttt{day} &= group_{24}(\texttt{hour})\\
\texttt{week} &= group_{7}(\texttt{day})\\
\texttt{busi-day} &= select-down_1^5(\texttt{day},\texttt{week})\\
\texttt{month} &= alter^{12*400}_{2+12*399,1}(\texttt{day}, alter^{12*100}_{2+12*99,-1}(\texttt{day}, alter^{12*4}_{2+12*3,1}(\texttt{day},\\
&\quad alter^{12}_{11,-1}(\texttt{day}, alter^{12}_{9,-1}(\texttt{day}, alter^{12}_{6,-1}(\texttt{day}, alter^{12}_{4,-1}(\texttt{day},\\
&\quad alter^{12}_{2,-3}(\texttt{day}, group_{31}(\texttt{day}))))))\\
\texttt{year} &= group_{12}(\texttt{month})\\
\texttt{academicyear} &= anchor-group(\texttt{day},\\
&\quad select-by-intersect_1^1(\texttt{busi-day}, select-down_9^1(\texttt{month})
\end{aligned}
$$

As a matter of fact, these granularities can be generated in a more controlled way. Indeed, the authors distinguish three layers of granularities:

L_1 containing the bottom granularity and all the granularities obtained by applying $group$, $alter$, and $shift$ on granularities of this layer;

L_2 including L_1 and containing all the granularities obtained by applying $subset$, $union$, $intersection$, and $difference$ on granularities of this layer and selections with first operand belonging to this layer;

L_3 including L_2 and containing all the granularities obtained by applying $combine$ on granularities of this layer and $anchor - group$ with the second operand on granularities of this layer.

Granularities of L_1 are full-integer labelled granularities, those of L_2 may not be labelled by all integers, but they contain no gaps within granules. These aspects, as well as the expressiveness of the generated granularities, are investigated in depth in [Bettini *et al.*, 2000].

3.3.5 Constraint solving and query answering

Wang et al. [Wang *et al.*, 1995] have proposed an extension of the relational data model which is able to handle granularity. The goal of this work is to take into account possible granularity mismatch in the context of federated databases.

An extended temporal model is a relational database in which each tuple is timestamped under some granularity. Formally, it is a set of tables such that each table is a quadruple

$\langle R, \phi, \tau, g\rangle$ such that R is a set of tuples (a relational table), g is a granularity, $\phi : \mathbb{N} \longrightarrow 2^R$ maps granules to tuples, $\tau : R \longrightarrow 2^{\mathbb{N}}$ maps tuples to granules such that $\forall t \in R, t \in \phi(i) \Rightarrow i \in \tau(t)$ and $\forall i \in \mathbb{N}, i \in \tau(t) \Rightarrow t \in \phi(i)$.

In [Bettini *et al.*, 2000], the authors develop methods for answering queries in database with granularities. The answers are computed with regard to hypotheses tied to the databases. These hypotheses allow the computation of values between two successive timestamps. The missing values can, for instance, be considered constant (persistence) or interpolated with a particular interpolation function. These hypotheses also apply to the computation of values between granularity.

The hypotheses (H) provide the way to compute the closure ($\overline{D}^H$) of a particular database (D). Answering a query q against a database with granularities D and hypotheses H consists in answering the query against the closure of the database ($\overline{D}^H \models q$). Instead of computing this costly closure, the authors proposes to $reduce$ the database with regard to the hypotheses (i.e., to find the minimal database equivalent to the initial one modulo closure) and to add to the query formulas allowing the computation of the hypotheses.

The authors also define quantitative temporal constraint satisfaction problems under granularity whose variables correspond to points and arcs are labelled by an integer interval and a granularity. A pair of points $\langle t, t'\rangle$ satisfies a constraint $[m, n]g$ (with $m, n \in \mathbb{Z}$ and g a granularity) if and only if $\uparrow^g t$ and $\uparrow^g t'$ are defined and $m \leq |\uparrow^g t - \uparrow^g t'| \leq n$. These constraints cannot be expressed as a classical TCSP (see Chapter 7). As a matter of fact, if the constraint $[0\ 0]$ is set on two entities under the hour granularity, two points satisfy it if they are in the same hour. In terms of seconds, the positions should differ from 0 to 3600. However, $[0\ 3600]$ under the second granularity does not corresponds to the original constraint since it can be satisfied by two points in different hours.

The satisfaction problem for granular constraint satisfaction is NP-hard (while STP is polynomial) [Bettini *et al.*, 1996]. Indeed the modulo operation involved in the conversions can introduce disjunctive constraints (or non convexity). For instance, next business day is the convex constraint ($[1\ 1]$), which converted in hours can yield the constraint $[1\ 24] \vee [49\ 72]$ which is dependent on the exact day of the week.

The authors propose an arc-consistency algorithm complete for consistency checking when the granularities are periodical with regard to some common finer granularity. They also propose an approximate (i.e., incomplete) algorithm by iterating the saturation of the networks of constraints expressed under the same granularity and then converting the new values into the other granularities.

The work described above mainly concerns aligned systems of granularity (i.e., systems in which the upward conversion is always defined). This is not always the case, as the week/month example illustrates it. Non-aligned granularity has been considered by several authors. Dyreson and collaborators [Dyreson and Snodgrass, 1994] define comparison operators across granularities and their semantics (this covers the extended comparators of [Wang *et al.*, 1995]): comparison between entities of different granularities can be considered under the coarser granularity (here coarser is the same as "groups into" above and thus requires alignment) or the finer one. They define upward and downward conversion operators across comparable granularities and the conversion across non-comparable granularities is carried out by first converting down to the greatest lower bound and then up (assuming the greatest lower bound exists and thus that the structure is a lower semi-lattice): $\downarrow^g_{g\wedge g'} \uparrow^{g'}_{g\wedge g'} x$. Comparisons across granularities (with both semantics) are implemented in terms of the conversion operators.

3.3.6 Alternative accounts of time granularity

The set-theoretic approach has been recently revisited and extended in several directions. In the following, we briefly summarize the most promising ones.

An alternative string-based model for time granularities has been proposed by Wijsen [Wijsen, 2000]. It models (infinite) granularities as (infinite) words over an alphabet consisting of three symbols, namely, ■ (filler), □ (gap), and ≀ (separator), which are respectively used to denote time points covered by some granule, to denote time points not covered by any granule, and to delimit granules. Wijsen focuses his attention on (infinite) periodical granularities, that is, granularities which are left bounded and, ultimately, periodically groups time points of the underlying temporal domain. Periodical granularities can be identified with ultimately periodic strings, and they can be finitely represented by specifying a (possibly empty) finite prefix and a finite repeating pattern. As an example, the granularity `BusinessWeek` ■■■■■□□ ≀ ■■■■■□□ ≀ . . . can be encoded by the empty prefix ε and the repeating pattern ■■■■■□□≀. Wijsen shows how to use the string-based model to solve some fundamental problems about granularities, such as the equivalence problem (to establish whether or not two given representations define the same granularity) and the minimization problem (to compute the most compact representation of a granularity). In particular, he provides a straightforward solution to the equivalence problem that takes advantage of a suitable *aligned form* of strings. Such a form forces separators to occur immediately after an occurrence of ■, thus guaranteeing a one-to-one correspondence between granularities and strings.

The idea of viewing time granularities as ultimately periodic strings establishes a natural connection with the field of formal languages and automata. An automaton-based approach to time granularity has been proposed by Dal Lago and Montanari in [Dal Lago and Montanari, 2001], and later revisited by Bresolin et al. in [Bresolin *et al.*, 2004; Dal Lago *et al.*, 2003a; Dal Lago *et al.*, 2003b]. The basic idea underlying such an approach is simple: we take an automaton $\mathcal{A}$ recognizing a *single* ultimately periodic word $u \in \{\square, \blacksquare, \blacktriangleleft\}^\omega$ and we say that $\mathcal{A}$ represents the granularity G if and only if u represents G. The resulting framework views granularities as strings generated by a specific class of automata, called Single-String Automata (SSA), thus making it possible to (re)use well-known results from automata theory. In order to compactly encode the redundancies of the temporal structures, SSA are endowed with counters ranging over discrete finite domains (Extended SSA, ESSA for short). Properties of ESSA have been exploited to efficiently solve the equivalence and the granule conversion problems for single time granularities [Dal Lago *et al.*, 2003b]. The relationships between ESSA and Calendar Algebra have been systematically investigated by Dal Lago et al. in [Dal Lago *et al.*, 2003a], where a number of algorithms that map Calendar Algebra expressions into automaton-based representations of time granularities are given. Such an encoding allows one to reduce problems about Calendar Algebra expressions to equivalent problems for ESSA. More generally, the operational flavor of ESSA suggests an alternative point of view on the role of automaton-based representations: besides a formalism for the direct specification of time granularities, automata can be viewed as a low-level formalism into which high-level time granularity specifications, such as those of Calendar Algebra, can be mapped. This allows one to exploit the benefits of both formalisms, using a high level language to define granularities and their properties in a natural and flexible way, and the automaton-based one to efficiently reason about them. Finally, a generalization of the automaton-based approach from single periodical granularities to (possibly infinite) sets of granularities has been proposed by Bresolin et al. in [Bresolin *et al.*, 2004]. To this end, they identify a proper subclass of Büchi automata, called Ultimately Periodic Automata (UPA), that captures regular sets consisting of only ultimately periodic words. UPA allow one to encode single granularities, (possibly infinite) sets of granularities which have the same repeating pattern and different prefixes, and sets of granularities characterized by a finite set of non-equivalent patterns, as well as any possible combination of them.

The choice of Propositional Linear Temporal Logic (Propositional LTL) as a logical tool for granularity management has been recently advocated by Combi et al. in [Combi *et al.*, 2004]. Time granularities are defined as models of Propositional LTL formulas, where suitable propositional symbols are used to mark the endpoints of granules. In this way, a large set of regular granularities, such as, for instance, repeating patterns that can start at an arbitrary time point, can be captured. Moreover, problems like checking the consistency of a granularity specification or the equivalence of two granularity expressions can be solved in a uniform way by reducing them to the validity problem for Propositional LTL, which is known to be in PSPACE. An extension of Propositional LTL that replaces propositional variables by first-order formulas defining integer constraints, e.g., $x \equiv_k y$, has been proposed by Demri in [Demri, 2004]. The resulting logic, denoted by PLTL^{mod}(Past LTL with integer periodicity constraints), generalizes both the logical framework proposed by Combi et al. and the automaton-based approach of Dal Lago and Montanari, and it allows one to compactly define granularities as periodicity constraints. In particular, the author shows how to reduce the equivalence problem for ESSA to the model checking problem for PLTL^{mod}(-automata), which turns out to be in PSPACE, as in the case of Propositional LTL. The logical approach to time granularity is systematically analyzed in the next section, where various temporal logics for time granularity are presented.

3.4 The logical approach

A first attempt at incorporating time granularity into a logical formalism is outlined in [Corsetti *et al.*, 1991a; Corsetti *et al.*, 1991b]. The proposed logical system for time granularity has two distinctive features. On the one hand, it extends the syntax of temporal logic to allow one to associate different granularities (temporal domains) with different subformulas of a given formula; on the other hand, it provides a set of translation rules to rewrite a subformula associated with a given granularity into a corresponding subformula associated with a finer granularity. In such a way, a model of a formula involving different granularities can be built by first translating everything to the finest granularity and then by interpreting the resulting (flat) formula in the standard way.

A major problem with such a method is that there exists no a standard way to define the meaning of a formula when moving from a time granularity to another one. Thus, more information is needed from the user to drive the translation of the (sub)formulas. The main idea is that when we state that a predicate p holds at a given time point x belonging to the temporal domain T, we mean that p holds in a subset of the interval corresponding to x in the finer domain T'. Such a subset can be the whole interval, a scattered sequence of smaller intervals, or even a single time point. For instance, saying that "the light has been switched on at time x_{min}", where x_{min} belong to the domain of minutes, may correspond to state that a predicate *switching_on* is true at the minute x_{min} and exactly at one second of x_{min}. Instead, saying that an employee works at the day x_d generally means that there are several minutes, during the day x_d, where the predicate *work* holds for the employee. These minutes are not necessarily contiguous. Thus, the logical system must provide the user with suitable tools that allow him to qualify the subset of time intervals of the finer temporal domain that correspond to the given time point of the coarser domain.

A substantially different approach is proposed in [Ciapessoni *et al.*, 1993; Montanari, 1994; Montanari, 1996], where Montanari et al. show how to extend syntax and semantics of temporal logic to cope with metric temporal properties possibly expressed at different time granularities. The resulting metric and layered temporal logic is described in detail in Subsection 3.4.1. Its distinctive feature is the coexistence of three different operators: a contextual operator, to associate different granularities with different (sub)formulas, a displacement operator, to move within a given granularity, and a projection operator, to move across granularities.

An alternative logical framework for time granularity has been developed in the classical logic setting [Montanari, 1996; Montanari and Policriti, 1996; Montanari *et al.*, 1999]. It imposes suitable restrictions to languages and structures for time granularity to get decidability. From a technical point of view, it defines various theories of time granularity as suitable extensions of monadic second-order theories of k successors, with $k \geq 1$. Monadic theories of time granularity are the subject of Subsection 3.4.2.

The temporal logic counterparts of the monadic theories of time granularity, called temporalized logics, are briefly presented in Subsection 3.4.3. This way back from the classical logic setting to the temporal logic one passes through an original class of automata, called temporalized automata.

A coda about the relationships between logics for time granularity and interval temporal logics concludes the section.

3.4.1 A metric and layered temporal logic for time granularity

Original metric and layered temporal logics for time granularity have been proposed by Montanari et al. in [Ciapessoni *et al.*, 1993; Montanari, 1994; Montanari, 1996]. We introduce these logics in two steps. First, we take into consideration their purely metric fragments in isolation. To do that, we adopt the general two-sorted framework proposed in [Montanari, 1996; Montanari and de Rijke, 1997], where a number of metric temporal logics, having a different expressive power, are defined as suitable combinations of a temporal component and an algebraic one. Successively, we show how flat metric temporal logic can be generalized to a many-layer metric temporal logic, embedding the notion of time granularity [Montanari, 1994; Montanari, 1996]. We first identify the main functionalities a logic for time granularity must support and the constraints it must satisfy; then, we axiomatically define metric and layered temporal logic, viewed as the combination of a number of differently-grained (single-layer) metric temporal logics, and we briefly discuss its logical properties.

The basic metric component

The idea of a logic of positions (topological, or metric, logic) was originally formulated by Rescher and Garson [Rescher and Garson, 1968; Rescher and Urquhart, 1971]. In [Rescher and Garson, 1968], the authors define the basic features of the logic and they show how to give it a temporal interpretation. Roughly speaking, metric (temporal) logic extends propositional logic with a parameterized operator Δ_α of positional realization that allows one to constrain the truth value of a proposition at position α. If we interpret the parameter α as a displacement with respect to the current position, which is left implicit, we have that $\Delta_\alpha q$ is true at a position x if and only if q is true at a position y at distance α from x. Metric temporal logics can thus be viewed as two-sorted logics having both formulas and parameters; formulas are evaluated at time points while parameters take values in a suitable algebraic structure of temporal displacements. In [Montanari and de Rijke, 1997], Montanari and de Rijke start with a very basic system of metric temporal logic, and they build on it by adding axioms and/or by enriching the underlying structures. In the following, we describe the metric temporal logic of two-sorted frames with a linear temporal order (*MTL*); we also briefly consider general metric temporal logics allowing quantification over algebraic and temporal variables and free mixing of algebraic and temporal formulas (*Q-MTL*).

The *two-sorted temporal language* for *MTL* has two components: the algebraic component and the temporal one. Given a non-empty set A of constants, let $T(A)$ be the set of terms over A, that is, the smallest set such that $A \subseteq T(A)$, and if α, $\beta \in T(A)$ then $\alpha + \beta$, $-\alpha, 0 \in T(A)$. The first-order (algebraic) component is built up from $T(A)$ and the predicate symbols $=$ and $<$. The temporal component of the language is built up from a non-empty set $\mathcal{P}$ of proposition letters. The set of formulas over $\mathcal{P}$ and A, $F(\mathcal{P}, A)$, is the smallest set such that $\mathcal{P} \subseteq F(\mathcal{P}, A)$, and if ϕ, $\psi \in F(\mathcal{P}, A)$ and $\alpha \in T(A)$, then $\neg\phi$, $\phi \wedge \psi$, $\top$ (true), $\bot$ (false), and $\Delta_\alpha\phi$ (and its dual $\nabla_\alpha\phi := \neg\Delta_\alpha\neg\phi$) belong to $F(\mathcal{P}, A)$. Δ_α is called the (parameterized) *displacement operator*.

A *two-sorted frame* is a triple $\mathbf{F} = (T, \mathbf{D}; \mathrm{DIS})$, where T is the set of (time) points over which temporal formulas are evaluated, $\mathbf{D}$ is the algebra of metric displacements in whose domain D terms take their values, and $\mathrm{DIS} \subseteq T \times D \times T$ is an accessibility relation, called displacement relation, relating pairs of points and displacements. The components of two-sorted frames satisfy the following properties. First, $\mathbf{D}$ is an ordered Abelian group, that is, a structure $\mathbf{D} = (D, +, -, 0, <)$, where $+$ is a binary function of displacement composition, $-$ is a unary function of inverse displacement, and 0 is the zero displacement constant, such that:

(i)	$\alpha + \beta = \beta + \alpha$	(commutativity of $+$);
(ii)	$\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$	(associativity of $+$);
(iii)	$\alpha + 0 = \alpha$	(zero element of $+$);
(iv)	$\alpha + (-\alpha) = 0$	(inverse),

and $<$ is an irreflexive, asymmetric, transitive, and linear relation that satisfies the comparability property (*viii*) below:

(v) $\neg(\alpha < \alpha)$;
(vi) $\neg(\alpha < \beta \wedge \beta < \alpha)$;
(vii) $\alpha < \beta \wedge \beta < \gamma \rightarrow \alpha < \gamma$;
$(viii)$ $\alpha < \beta \vee \alpha = \beta \vee \beta < \alpha$.

Furthermore, there are two conditions expressing the relations between $+$ and $-$, and $<$:

(ix) $\alpha < \beta \rightarrow \alpha + \gamma < \beta + \gamma$;
(x) $\alpha < \beta \rightarrow -\beta < -\alpha$.

As for the displacement relation, we first require DIS to respect the converse operation of the Abelian group in the following sense:

Symmetry: $\forall i, j, \alpha\, (\mathrm{DIS}(i, \alpha, j) \rightarrow \mathrm{DIS}(j, -\alpha, i))$.

Furthermore, we require DIS to be reflexive, transitive, quasi-functional (q-functional for short) with respect to both its third and second argument, and totally connected:

Reflexivity: $\forall i\, \mathrm{DIS}(i, 0, i)$;
Transitivity: $\forall i, j, k, \alpha, \beta\, (\mathrm{DIS}(i, \alpha, j) \wedge \mathrm{DIS}(j, \beta, k) \rightarrow \mathrm{DIS}(i, \alpha + \beta, k))$;
Q-functionality - 1: $\forall i, j, j', \alpha\, (\mathrm{DIS}(i, \alpha, j) \wedge \mathrm{DIS}(i, \alpha, j') \rightarrow j = j')$;
Q-functionality - 2: $\forall i, j, \alpha, \beta\, (\mathrm{DIS}(i, \alpha, j) \wedge \mathrm{DIS}(i, \beta, j) \rightarrow \alpha = \beta)$;
Total connectedness: $\forall i, j \exists \alpha\, \mathrm{DIS}(i, \alpha, j)$.

From the ordering $<$ on the algebraic component of the frames, an ordering $\ll$ on the temporal component can be defined as follows:

$$i \ll j \text{ iff for some } \alpha > 0, \mathrm{DIS}(i, \alpha, j). \tag{3.1}$$

According to Definition 3.1, we have that i and j are $\ll$-related if there exists a positive displacement between them. It is possible to show that $\ll$ is a strict linear order [Montanari and de Rijke, 1997] (it is worth noting that, without the properties of quasi-functionality with respect to the second argument and total connectedness, Definition 3.1 does not produce a strict linear order).

The interpretation of the language for *MTL* on two-sorted frames based on an ordered Abelian group is fairly straightforward: the first-order (algebraic) component is interpreted on the ordered Abelian group, and the temporal component on the temporal domain. Basically, a two-sorted frame **F** can be turned into a *two-sorted model* by adding an interpretation for the algebraic terms and a valuation for proposition letters. An interpretation for algebraic terms is given by a function $g : A \rightarrow D$ that is automatically extended to all terms from $T(A)$. A valuation is simply a function $V : \mathcal{P} \rightarrow 2^T$. We say that $\alpha = \beta$ (resp. $\alpha < \beta$) is *true* in a model $\mathbf{M} = (T, \mathbf{D}; \mathrm{DIS}; V, g)$ whenever $g(\alpha) = g(\beta)$ (resp. $g(\alpha) < g(\beta)$). *Truth* of temporal formulas is defined by means of the standard semantic clauses for proposition letters and Boolean connectives, plus the following clause for the displacement operator:

$$\mathbf{M}, i \Vdash \Delta_\alpha \phi \quad \text{iff} \quad \text{there exists } j \text{ such that } \mathrm{DIS}(i, g(\alpha), j) \text{ and } \mathbf{M}, j \Vdash \phi.$$

Let Γ denote a set of formulas. To avoid messy complications we only consider one-sorted consequences $\Gamma \models \phi$; for algebraic formulas '$\Gamma \models \phi$' means 'for all models $\mathbf{M}$, if $\mathbf{M} \models \Gamma$, then $\mathbf{M} \models \phi$'; for temporal formulas it means 'for all models $\mathbf{M}$, and time points i, if $\mathbf{M}, i \Vdash \Gamma$, then $\mathbf{M}, i \Vdash \phi$'.

The following example shows that the language of *MTL* allows one to express meaningful temporal conditions.

Example 3.4.1. *Let us consider a communication channel C that collects messages from n different sources $S_1, \ldots, S_n$ and outputs them with delay δ. To exclude that two input events can occur simultaneously, we add the constraint (notice that preventing input events from occurring simultaneously also guarantees that output events do not occur simultaneously):*

$$\forall i, j\, \neg(\mathtt{in}(i) \wedge \mathtt{in}(j) \wedge i \neq j),$$

which is shorthand for:

$$\neg(\mathtt{in}(1) \wedge \mathtt{in}(2)) \wedge \ldots \wedge \neg(\mathtt{in}(n-1) \wedge \mathtt{in}(n)).$$

The behavior of C is specified by the formula:

$$\forall i\,(\mathtt{out}(i) \leftrightarrow \Delta_{-\delta}\mathtt{in}(i)),$$

which is shorthand for a finite conjunction.

Validity in *MTL* can be axiomatized as follows. For the displacement component, one takes the axioms and rules of identity, ordered Abelian groups, and strict linear order, together with any complete calculus for first-order logic. For the temporal component, one takes the usual axioms of propositional logic plus the axioms:

(AxND)	$\nabla_\alpha(p \to q) \to (\nabla_\alpha p \to \nabla_\alpha q)$	(normality);
(AxSD)	$p \to \nabla_\alpha \Delta_{-\alpha} p,$	(symmetry);
(AxRD)	$\nabla_0 p \to p$	(reflexivity);
(AxTD)	$\nabla_{\alpha+\beta} p \to \nabla_\alpha \nabla_\beta p$	(transitivity);
(AxQD)	$\Delta_\alpha p \to \nabla_\alpha p$	(q-functionality - 1).

Its rules are modus ponens and

(D-NEC)	$\vdash \phi \implies \vdash \nabla_\alpha \phi$	(necessitation rule for ∇_α);
(REP)	$\vdash \phi \leftrightarrow \psi \implies \vdash \chi(\phi/p) \leftrightarrow \chi(\psi/p)$	(replacement),
	where (ϕ/p) denotes substitution of ϕ for the variable p;	
(LIFT)	$\vdash \alpha = \beta \implies \vdash \nabla_\alpha \phi \leftrightarrow \nabla_\beta \phi$	(transfer of identities).

Axiom (AxN) is the usual distribution axiom; axiom (AxS) expresses that a displacement α is the converse of a displacement $-\alpha$; axioms (AxR), (AxT), and (AxQ) capture reflexivity, transitivity, and quasi-functionality with respect to the third argument, respectively. A suitable adaptation of two truth preserving constructions from standard modal logic to the *MTL* setting allows one to show there are no *MTL* formulas that express total connectedness and quasi-functionality with respect to the second argument of the displacement relation [Montanari and de Rijke, 1997]. The rules (D-NEC) and (REP) are familiar from modal logic. Finally, the rule (LIFT) allows one to transfer provable algebraic identities from the displacement domain to the temporal one.

A *derivation in MTL* is a sequence of formulas $\sigma_1, \ldots, \sigma_n$ such that each σ_i, with $1 \leq i \leq n$, is either an axiom or obtained from $\sigma_1, \ldots, \sigma_{n-1}$ by applying one of the derivation rules of *MTL*. We write $\vdash_{MTL} \sigma$ to denote that there is a derivation in *MTL* that ends in σ. It immediately follows that $\vdash_{MTL} \alpha = \beta$ iff $\alpha = \beta$ is provable from the axioms of the algebraic component only: whereas we can lift algebraic information from the displacement domain to the temporal domain using the (LIFT) rule, there is no way in which we can import temporal information into the displacement domain. As with consequences, we only consider one-sorted inferences '$\Gamma \vdash \phi$'.

Theorem 3.4.1. *MTL is sound and complete for the class of all transitive, reflexive, totally-connected, and quasi-functional (in both the second and third argument of their displacement relation) frames.*

The proof of soundness is trivial. The completeness proof is much more involved [Montanari and de Rijke, 1997]. It is accomplished in two steps: first, one proves completeness with respect to totally connected frames via same sort of generated submodel construction; then, a second construction is needed to guarantee quasi-functionality with respect to the second argument.

Propositional variants of *MTL* are studied in [Montanari and de Rijke, 1997]. As an example, one natural specialization of *MTL* is obtained by adding discreteness. As in the case of the ordering, the discreteness of the temporal domain necessarily follows from that of the domain of temporal displacements, which is expressed by the following formula:

$$\forall\alpha\,\exists\beta,\beta^{'}(\alpha<\beta\wedge\forall\gamma(\alpha<\gamma\rightarrow(\beta=\gamma\vee\beta<\gamma))\wedge$$
$$\beta^{'}<\alpha\wedge\forall\delta(\delta<\alpha\rightarrow(\beta^{'}=\delta\vee\beta^{'}<\delta)))$$

Proposition 3.4.1. *Let* $\mathbf{F}=(T,\mathbf{D};\mathrm{DIS})$ *be a two-sorted frame based on a discrete ordered Abelian group* $\mathbf{D}$*. For all* $i,j\in T$*, there exist only finitely many* k *such that* $i\ll k\ll j$*.*

For some applications, both *MTL* and its propositional variants are not expressive enough, and thus they must be extended. In particular, they lack quantification and constrain displacements to occur as parameters of the displacement operator only. The following example shows how the ability of freely mixing temporal and displacement formulas enables one to exploit more complex ways of interaction between the two domains, rather than to only lift information from the algebraic domain to the temporal one.

Example 3.4.2. *Let us consider the operation of a traffic light controller* C *[Henzinger* et al.*, 1994]. When the request button is pushed, the controller makes a pedestrian light turn green within a given time bound after which the light remains green for a certain amount of time. Moreover, assume that* C *takes a unit of time to switch the light and that the time needed for its internal operations is negligible.*

We require that C *satisfies the following conditions:*

(i) whenever a pedestrian pushes the request button ('request is true'), then the light is green within 5 time units and remains green for at least 10 time units (this condition guarantees that no pedestrian waits for more than 5 time units, and that he or she is given at least 10 time units to cross the road);

(ii) whenever request is true, then it is false within 20 time units (this condition ensures that the request button is reset);

(iii) whenever request has been false for 20 time units, the light is red (this condition should prevent the light from always being green).

By taking advantage of the possibility of quantifying displacement variables and of using displacement formulas, the behavior of C *can be specified by the conjunction of the following formulas:*

$$\text{request} \quad\rightarrow\quad \exists x(0<x\leq 5\wedge\forall y(x\leq y<x+10\rightarrow\nabla_y\text{lightIsGreen}));$$
$$\text{request} \quad\rightarrow\quad \exists z(0\leq z\leq 20\wedge\Delta_z\neg\,\text{request});$$
$$\forall x(0\leq x<20\rightarrow\nabla_x\neg\text{request})\rightarrow\nabla_{20}\text{lightIsRed},$$

together with a formula stating that at each time point the traffic light is either red or green:

$$\text{lightIsGreen} \leftrightarrow \neg\ \text{lightIsRed}.$$

Different implementations of C, all satisfying the given specification, can be obtained by making different assumptions about the value of temporal parameters, e.g., by varying the delay between requests and resets. It is worth noting that, even if there are no restrictions on the frequency of requests, the above specification is appropriate only if that frequency is low; otherwise, it may happen that switching the light to red is delayed indefinitely. A solution to this problem is discussed in [Montanari, 1996].

Systems of quantified metric temporal logic (*Q-MTL* for short) are developed in [Montanari and de Rijke, 1997]. The language of *Q-MTL* extends that of *MTL* by adding algebraic variables (and, possibly, temporal variables) and by allowing quantification over algebraic (and temporal) variables and free mixing of algebraic formulas and temporal propositional symbols. *Q-MTL* models can be obtained from ordered two-sorted frames $\mathbf{F} = (T, \mathbf{D}; \mathrm{DIS})$ by adding an interpretation function g for the algebraic terms and a valuation V for proposition letters, and by specifying the way one evaluates mixed formulas at time points. An axiomatic system for *Q-MTL* (we refer to the simplest system of quantified metric temporal logic; other cases are considered in [Montanari and de Rijke, 1997]) is obtained from that for *MTL* by adding a number of axiom schemata governing the behavior of quantifiers and substitutions:

(AxF) $\forall x\,(\phi \rightarrow \psi) \leftrightarrow (\forall x\,\phi \rightarrow \forall x\,\psi)$ (functionality);
(AxEVQ) $\phi \rightarrow \forall x\,\phi$, for x not in ϕ
(elimination of vacuous quantifiers);
(AxUI) $\forall x\,\phi \rightarrow \phi(\alpha/x)$, with α free for x in ϕ
(universal instantiation),

the Barcan formula for the displacement operator:

(AxBFD) $\forall x\,\nabla_\alpha \phi \rightarrow \nabla_\alpha \forall x\,\phi$, with $x \notin \alpha$ (Barcan formula for ∇_α),
where $x \notin \alpha$ stands for $x \neq \alpha$ and x does not occur in α,

the axioms relating the algebraic terms and the displacement operator (axiom (AxAD4) can actually be derived from the other axioms):

(AxAD1) $\alpha = \beta \rightarrow \forall x \nabla_x \alpha = \beta$; (AxAD2) $\alpha \neq \beta \rightarrow \forall x \nabla_x \alpha \neq \beta$;
(AxAD3) $\alpha < \beta \rightarrow \forall x \nabla_x \alpha < \beta$; (AxAD4) $\alpha \not< \beta \rightarrow \forall x \nabla_x \alpha \not< \beta$,

and the rule:

(UG) $\vdash \phi \implies \vdash \forall x\,\phi$ (universal generalization).

The completeness of *Q-MTL* can be proved by following the general pattern of the completeness proof for *MTL*, but the presence of mixed formulas complicates some of the details. Basically, it makes use of a variant of Hughes and Cresswell's method for proving axiomatic completeness in the presence of the Barcan formula [Hughes and Cresswell, 1968].

The addition of time granularity

Metric and Layered Temporal Logic (*MLTL* for short) is obtained from *MTL* by adding a notion of time granularity [Ciapessoni *et al.*, 1993; Montanari, 1994; Montanari, 1996]. In the following, we first show how to extend two-sorted frames to incorporate granularity; then, we present syntax, semantics, and axiomatization of *MLTL*; finally, we briefly describe the way in which the synchronization problem (cf. Section 3.2) can be dealt with in *MLTL*.

The main change to make to the model of time when moving from *MTL* to *MLTL* is the replacement of the temporal domain T by a temporal universe $\mathcal{T}$ consisting of a set of *disjoint* linear temporal domains/layers, that share the same displacement domain D. Formally, $\mathcal{T} = \{T^i : i \in M\}$, where M is an initial segment of $\mathbb{N}$, possibly equal to $\mathbb{N}$. The set $\bigcup_{i \in M} T^i$ collects all time points belonging to the different layers of $\mathcal{T}$. $\mathcal{T}$ is assumed to be *totally ordered* by the granularity relation $\prec$. As an example, if $\mathcal{T} = \{\texttt{years}, \texttt{months}, \texttt{weeks}, \texttt{days}\}$, we have that $\texttt{days} \prec \texttt{weeks} \prec \texttt{months} \prec \texttt{years}$. A finer characterization of the relations among the layers of a temporal universe is provided by the *disjointedness* relation, denoted by $\subset$, which is quite similar to the *groups-into* relation defined in Section 3.3. It defines a partial order over $\mathcal{T}$ that rules out pairs of layers like `weeks` and `months` for which a point of a finer layer (`weeks`) can be astride two points of the coarser one (`months`). As an example, given $\mathcal{T} = \{\texttt{years}, \texttt{months}, \texttt{weeks}, \texttt{days}\}$, we have that $\texttt{months} \subset \texttt{years}, \texttt{days} \subset \texttt{months}$, and $\texttt{days} \subset \texttt{weeks}$. This means that `years` are pairwise disjoint when viewed as sets of `months`; the same holds for `months` when viewed as sets of `days`.

The links between points belonging to the same layer are expressed by means of (a number of instances of) the *displacement* relation, while those between points belonging to different layers are given by means of a *decomposition* relation that, for every pair $T^i, T^j \in \mathcal{T}$, with $T^j \prec T^i$, associates each point of T^i with the set of points of T^j that compose it. We assume that the decomposition relation turns every point $x \in T^i$ into a set of contiguous points (decomposition interval) of T^j (*convexity*). This condition excludes the presence of 'temporal gaps' within the set of components of a given point, as it happens, for instance, when `business months` are mapped on `days`. In general, the cardinalities of the sets of components of two distinct points $x, y \in T^i$ with respect to T^j may be different (*non homogeneity*). This is the case, for instance, with pairs of layers like `real months` and `days`: different months can be mapped on a different number of days ($28, 29, 30$, or 31). In some particular contexts, however, it is convenient to work with temporal universes where, for every pair of layers T^i, T_j, with $T^j \prec T^i$, the decomposition intervals have the same cardinality (*homogeneity*). For instance, this is the case of temporal universes that replace `real months` by `legal months`, which, conventionally, are 30-days long. We constrain the decomposition relation to respect the ordering of points within layers (*order preservation*). If $T^j \subset T^i$, e.g., `seconds` and `minutes`, then the intervals are disjoint; otherwise, the intervals can possibly meet at their endpoints, e.g., `weeks` and `months`. We further require that the union of the intervals of T^j associated with the points of T^i covers the whole T^j (*total covering*). From order preservation and total covering, it follows that, for all pairs of layers T^i, T^j, with $T^j \prec T^i$, the decomposition relation associates contiguous points of T^i with contiguous sets of points of T_j (*contiguity*). This excludes the presence of 'temporal gaps' between the decomposition intervals of consecutive points of the coarser layer, as it happens, for instance, when `business weeks` are mapped on `days`. Finally, we require that, for every i, j, k, if $T^j \subset T^k \subset T^i$, then the decomposition of T^i into T^j can be obtained from the decomposition of T^i into T^k and that of T^k into T^j (*downward transitivity*). The same holds for $T^k \subset T^j \subset T^i$ (*downward/upward transitivity*). In the following, we shall also consider the inverse relation of *abstraction*, that, for every pair $T^i, T^j \in \mathcal{T}$, with

$T^j \prec T^i$, associates a point $x \in T^j$ with a point $y \in T^i$ if x belongs to the the decomposition of y with respect to T^j. Every point $x \in T^j$ can be abstracted into either one or two adjacent points of T^i. If $T^j \subset T^i$, x is abstracted into a unique point y, which is called the *coarse grain equivalent* of x with respect to T^i.

Besides the algebraic and temporal components, the *temporal language* for *MLTL* includes a *context sort*. Moreover, the displacement operator is paired with a contextual operator and a projection operator. Formally, given a non-empty set C of context constants, denoting the layers of the temporal universe, and a set Y of context variables, the set $T(C \cup Y)$ of context terms is equal to $C \cup Y$. The set $T(A \cup X)$ of algebraic terms denoting temporal displacements is built up as follows. Let A be a set of algebraic constants and X be a set of algebraic variables. $T(A \cup X)$ is the smallest set such that $A \subseteq T(A \cup X)$, $X \subseteq T(A \cup X)$, and if α, $\beta \in T(A \cup X)$ then $\alpha + \beta$, $-\alpha, 0 \in T(A \cup X)$. Finally, given a non-empty set of proposition letters $\mathcal{P}$, the set of formulas $F(\mathcal{P}, A, X, C, Y)$ is the smallest set such that $\mathcal{P} \in F(\mathcal{P}, A, X, C, Y)$, if $\phi, \psi \in F(\mathcal{P}, A, X, C, Y)$, $x \in X$, $y \in Y$, $c, c', c'' \in T(C \cup Y)$, and $\alpha, \beta \in T(X \cup A)$, then $\alpha = \beta, \alpha < \beta, c' \prec c'', c' \subset c'', \neg\phi, \phi \wedge \phi, \Delta_\alpha \phi$ (and $\nabla_\alpha \phi$), $\Delta^c \phi$ (and its dual $\nabla^c \phi := \neg\Delta^c\neg\phi$), $\Diamond\phi$ (and its dual $\Box\phi := \neg\Diamond\neg\phi$), $\forall x\phi$, and $\forall y\phi$ belong to $F(\mathcal{P}, A, X, C, Y)$. Δ^c is called the (parameterized) *contextual operator*. When applied to a formula ϕ, it restricts the evaluation of ϕ to the time points of the layer denoted by c. The combined use of Δ_α and Δ^c makes it possible to define a derived operator Δ^c_α of *contextualized* (or *local*) *displacement*: $\Delta^c_\alpha \phi := \Delta^c \Delta_\alpha \phi$ (and its dual $\nabla^c_\alpha \phi := \nabla^c \nabla_\alpha \phi$). In such a case, the context term c can be viewed as the sort of the algebraic term α (*multi-sorted algebraic terms*). $\Diamond$ is called the *projection operator*. When applied to a formula ϕ, it allows one to evaluate ϕ at time points which are descendants (decomposition) or ancestors (abstraction) of the current one. Restrictions to specific sets of descendants or ancestors can be obtained by pairing the projection operator with the contextual one.

The *two-sorted frame* for time granularity is a tuple

$$\mathbf{F} = ((\mathcal{T}, \prec, \subset), \mathbf{D};\ \mathrm{DIS}, \mathrm{CONT}, \updownarrow)$$

where $\mathcal{T}$ is the temporal universe, $\prec$ and $\subset$ are the granularity and disjointedness relations, respectively, $\mathbf{D}$ is the algebra of metric displacements, $\mathrm{DIS} = \bigcup_{i \in M} \mathrm{DIS}_i$ is the displacement relation, $\mathrm{CONT} \subseteq \bigcup_{i \in M} T^i \times \mathcal{T}$ is the relation of contextualization, and $\updownarrow \subseteq \bigcup_{i \in M} T^i \times \bigcup_{i \in M} T^i$ is the projection relation. $\mathcal{T}$ is totally (resp. partially) ordered by $\prec$ (resp. $\subset$). For every layer T^i, the ternary relation $\mathrm{DIS}_i \subseteq T^i \times D \times T^i$ relates pairs of time points in T^i to a displacement in D. We assume that all DIS_i satisfy the same properties. The relation CONT associates each time point with the layer it belongs to. In its full generality, such a relation allows one point to belong to more than one layer (overlapping layers). However, since we restricted ourselves to the case in which $\mathcal{T}$ is totally ordered by "$\prec$', we assume that $\mathcal{T}$ defines a partition of $\bigcup_{i \in M} T^i$. This amounts to constrain CONT to be a total function with range equal to $\mathcal{T}$. The projection relation $\updownarrow$ associates each point with its direct or indirect descendants (downward projection) and ancestors (upward projection). More precisely, for any pair of points x, y, $\updownarrow (x, y)$ means that either x downward projects on y or x upward projection on y. Different temporal structures for time granularity can be obtained by imposing different conditions on the projection relation. Here is the list of the basic properties of the projection relation, where we assume variables x, y, z to take value over (subsets of) $\bigcup_{i \in M} T^i$ and variables α, β to take value over D:

reflexivity every point x projects on itself

$$\forall x \ \updownarrow(x,x)$$

uniqueness the projection relation does not link distinct points belonging to the same layer

$$\forall x, y, T^i((x \in T^i \wedge y \in T^i \wedge x \neq y) \rightarrow \neg \updownarrow(x,y))$$

refinement - case 1 for any pair of layers T^i, T^j, with $T^j \prec T^i$, any point of T^i projects on at least two points of T^j

$$\forall T^i, T^j, x \ \exists y, z((T^j \prec T^i \wedge x \in T^i) \rightarrow \\ (y \in T^j \wedge z \in T^j \wedge y \neq z \wedge \updownarrow(x,y) \wedge \updownarrow(x,z)))$$

refinement - case 2 for any pair of layers T^i, T^j, with $T^j \prec T^i$, and every point $x \in T^i$, there exists at least one point $y \in T^j$ such that x projects on y and no other point $z \in T^i$ projects on it

$$\forall T^i, T^j, x \ \exists y((T^j \prec T^i \wedge x \in T^i) \rightarrow (y \in T^j \wedge \\ \updownarrow(x,y) \wedge \forall z((z \in T^i \wedge z \neq x) \rightarrow \neg \updownarrow(z,y))))$$

separation for any pair of layers T^i, T^j, with $T^j \subset T^i$, the decomposition intervals of distinct points of T^i are disjoint

$$\forall T^i, T^j, x, y, x', y'((T^j \subset T^i \wedge x \in T^i \wedge y \in T^i \wedge x \neq y \wedge \\ x' \in T^j \wedge y' \in T^j \wedge \updownarrow(x,x') \wedge \updownarrow(y,y')) \rightarrow x' \neq y')$$

symmetry if x downward (resp. upward) projects on y, then y upward (resp. downward) projects on x

$$\forall x, y(\updownarrow(x,y) \rightarrow \updownarrow(y,x))$$

By pairing symmetry and separation, it easily follows that, whenever $T^j \subset T^i$, each point of the finer layer is projected on a unique point of the coarser one (*alignment*).
downward transitivity if $T^k \subset T^j \subset T^i$, $x \in T^i$ projects on $y \in T^j$, and y projects on $z \in T^k$, then x projects on z

$$\forall T^i, T^j, T^k, x, y, z((T^k \subset T^j \subset T^i \wedge x \in T^i \wedge \\ y \in T^j \wedge z \in T^k \wedge \updownarrow(x,y) \wedge \updownarrow(y,z)) \rightarrow \updownarrow(x,z))$$

Notice that we cannot substitute $\prec$ for $\subset$ in the above formula. Consider a temporal universe consisting of `months`, `weeks`, and `days`. The week from December 29, 2003, to January 4, 2004, belongs to the decomposition of December 2003 (as well as of January 2004) and the 3rd of January 2003 belongs to the decomposition of such a week, but not to that of December 2003.
downward/upward transitivity - case 1 if $T^j \subset T^k \subset T^i$, $x \in T^i$ projects on $y \in T^j$, and y projects on $z \in T^k$, then x projects on z

$$\forall T^i, T^j, T^k, x, y, z((T^j \subset T^k \subset T^i \wedge x \in T^i \wedge \\ y \in T^j \wedge z \in T^k \wedge \updownarrow(x,y) \wedge \updownarrow(y,z)) \rightarrow \updownarrow(x,z))$$

As in the case of downward transitivity, we cannot substitute $\prec$ for $\subset$ in the above formula. Consider a temporal universe consisting of `years`, `months`, and `weeks`. The week from December 29, 2003, to January 4, 2004, belongs both to the decomposition of the year 2003 (as well as of the year 2004) and to the decomposition of the month of January 2004, but such a month does not belong to the decomposition of the year 2003.

order preservation the linear order of layers is preserved by the projection relation. For every pair T^i, T^j, the projection intervals are ordered, but they can possibly meet (weak order preservation)

$$\forall T^i, T^j, x, y, x', y'((x \in T^i \wedge y \in T^i \wedge x' \in T^j \wedge y' \in T^j \wedge \\ \updownarrow(x, x') \wedge \updownarrow(y, y') \wedge x \ll y) \rightarrow (x' \ll y' \vee x' = y'))$$

where $x \ll y$ iff for some $i \in M$ and $\alpha > 0$, $\mathrm{DIS}_i(x, \alpha, y)$. Weak order preservation encompasses both the case of two months that share a week and the case of two months that belong to the same year.

From refinement (cases 1 and 2), symmetry and weak order preservation, it follows that, for any pair of layers T^i, T^j, with $T^j \prec T^i$, any point of T^j projects on either one or two points of T^i (*abstraction*). Moreover, from refinement (case 2), symmetry, and weak order preservation, it follows that it is never the case that, given any pair of layers T^i, T^j, with $T^j \prec T^i$, two consecutive points of T^j are both projected on the same two points of T^i.

If $T^j \subset T^i$, the projection intervals of the elements of T^i over T^j are ordered and disjoint, that is, we must substitute $x' \ll y'$ for $x' \ll y' \vee x' = y'$ (*strong order preservation*).

convexity for any ordered pair of layers T^i, T^j (either $T^i \prec T^j$ or $T^j \prec T^i$), the projection relation associates any point of T^i with an interval of contiguous points of T^j

$$\forall T^i, T^j, x, y, w, z((x \in T^i \wedge y \in T^j \wedge z \in T^j \wedge w \in T^j \wedge \\ y \ll w \wedge w \ll z \wedge \updownarrow(x, y) \wedge \updownarrow(x, z)) \rightarrow \updownarrow(x, w))$$

In some situations, the layers of the temporal universe can be assumed to (pairwise) satisfy the property of homogeneity.

homogeneity for every pair of (discrete) layers ordered by granularity, the projection relation associates the same number of points of the finer layer with every point of the coarser one

$$\forall T^i, T^j, x, y, x', x'' \exists y', y''((T^j \prec T^i \wedge x \in T^i \wedge y \in T^i \wedge x' \in T^j \wedge x'' \in T^j \wedge \\ x' \neq x'' \wedge \updownarrow(x, x') \wedge \updownarrow(x, x'')) \rightarrow (y' \in T^j \wedge y'' \in T^j \wedge y' \neq y'' \wedge \updownarrow(y, y') \wedge \updownarrow(y, y'')))$$

and

$$\forall T^i, T^j, x, y, y' \exists x'((T^j \prec T^i \wedge x \in T^i \wedge y \in T^i \wedge \\ y' \in T^j \wedge \updownarrow(y, y')) \rightarrow (x' \in T^j \wedge \updownarrow(x, x')))$$

Other interesting properties of the projection relation can be derived from the above ones, including *total covering*, *contiguity*, *seriality* (any point x can be projected on any layer T^i), *upward transitivity* (if $T^k \subset T^j \subset T^i$, $x \in T^k$ projects on $y \in T^j$, and y projects on $z \in T^i$, then x projects on z), and *downward/upward transitivity - case 2* (if $T^j \subset T^i \subset T^k$, $x \in T^i$ projects on $y \in T^j$, and y projects on $z \in T^k$, then x projects on z).

To turn a two-sorted frame $\mathbf{F}$ into a *two-sorted model* $\mathbf{M}$, we first add the interpretations for context and algebraic terms, and the valuation for atomic temporal formulas. The interpretation for context terms is given by a function $h : C \cup Y \rightarrow \mathcal{T}$; that for algebraic terms is given by a function $g : A \cup X \to D$, which is automatically extended to all terms from $T(A \cup X)$. The valuation V for propositional variables is defined as in *MTL*. An atomic formula of the form $\alpha = \beta$ (resp. $\alpha < \beta$) is *true* in a model $\mathbf{M} = (\mathbf{F}; V, g, h)$ whenever $g(\alpha) = g(\beta)$ (resp. $g(\alpha) < g(\beta)$). Analogously, $c \prec c'$ (resp. $c \subset c'$) is *true* in $\mathbf{M}$ whenever $h(c) \prec h(c')$ (resp. $h(c) \subset h(c')$). Next, the *truth* of the temporal formulas $\Delta_\alpha \phi$, $\Delta^c \phi$, and $\Diamond \phi$ is defined by the following clauses:

$$
\begin{array}{rll}
\mathbf{M}, i \Vdash \Delta_\alpha \phi & \text{iff} & \text{there exists } j \text{ such that } \mathrm{DIS}(i, g(\alpha), j) \text{ and } \mathbf{M}, j \Vdash \phi; \\
\mathbf{M}, i \Vdash \Delta^c \phi & \text{iff} & \mathrm{CONT}(i, h(c)) \text{ and } \mathbf{M}, i \Vdash \phi; \\
\mathbf{M}, i \Vdash \Diamond \phi & \text{iff} & \text{there exists } j \text{ such that } \updownarrow (i, j) \text{ and } \mathbf{M}, j \Vdash \phi.
\end{array}
$$

The semantic clauses for the dual operators ∇_α, ∇^c, and $\Diamond$, as well as for the derived operator Δ^c_α, can be easily derived from the above ones. Note that $\Delta^c \phi$ (resp. $\nabla^c \phi$) conventionally evaluates to false (resp. true) outside the context c. Finally, to evaluate the quantified formula $\forall x\, \phi$, with $x \in X$ (resp. $\forall y\, \phi$, with $y \in Y$), at a point i, we write $g =_x g'$ (resp. $h =_y h'$) to state that the assignments g and g' (resp. h and h') agree on all variables except maybe x (resp. y). We have that $(\mathbf{F}; V, g, h), i \Vdash \forall x\, \phi$ iff $(\mathbf{F}; V, g', h), i \Vdash \phi$, for all assignments g' such that $g =_x g'$. Analogously for $\forall y\, \phi$.

The notions of satisfiability, validity, and logical consequence given for *MTL* can be easily generalized to *MLTL*. Furthermore, the layered structure of *MLTL*-frames makes it possible to define the notions of *local* satisfiability, validity, and logical consequence by restricting the general notions of satisfiability, validity, and logical consequence to a specific layer.

The following examples show how *MLTL* allows one to specify temporal conditions involving different time granularities (the application of *MLTL* to the specification of complex real-time systems is discussed in [Montanari, 1996]). In the simplest case (case (i)), *MLTL* specifications are obtained by contextualizing formulas and composing them by means of logical connectives. The projection operator is needed when displacements over different layers have to be composed (case (ii)). Finally, contextual and projection operators can be paired to specify nested quantifications (cases (iii)-(vi)).

Example 3.4.3. *Consider the temporal conditions expressed by the following sentences:*

(i) men work every month and eat every day;

(ii) in 20 seconds 5 minutes will have passed from the occurrence of the fault;

(iii) some days the plant works every hour;

(iv) some days the plant remains inactive for several hours;

(v) every day the plant is in production for some hours;

(vi) the plant is monitored by the remote system every minute of every hour.

They can be expressed in MLTL *by means of the following formulas:*

(i) $\forall x_{man}(\forall \alpha \nabla^{month}_{\alpha} \mathtt{work}(x_{man}) \wedge \forall \beta \nabla^{day}_{\beta} \mathtt{eat}(x_{man}))$;

(ii) $\Delta^{second}_{20} \Diamond \Delta^{minute}_{-5} \mathtt{fault}$;

(iii) $\exists\alpha\Delta_\alpha^{day}\Box\nabla^{hour}\mathtt{work}(\mathtt{plant})$;

(iv) $\exists\alpha\Delta_\alpha^{day}\Diamond\Delta^{hour}\mathtt{inactive}(\mathtt{plant})$;

(v) $\forall\alpha\nabla_\alpha^{day}\Diamond\Delta^{hour}\mathtt{in\_production}(\mathtt{plant})$;

(vi) $\forall\alpha\nabla_\alpha^{hour}\Box\nabla^{minute}\mathtt{monitor}(\mathtt{remote\text{-}system}, \mathtt{plant})$.

As a matter of fact, it is possible to give a stronger interpretation of condition (ii), which is expressed by the formula:

(ii') $\Delta_{20}^{second}\Diamond\Delta_{-5}^{minute}\mathtt{fault} \wedge \forall\alpha(0 \leq \alpha < 20 \rightarrow \neg\Delta_\alpha^{second}\Diamond\Delta_{-5}^{minute}\mathtt{fault})$.

The problem of finding an axiomatization of validity in *MLTL* is addressed in [Ciapessoni *et al.*, 1993; Montanari, 1996]. The idea is to pair axioms and rules of *(Q-)MTL*, which are used to express the properties of the displacement operator with respect to every context, with additional axiom schemata and rules governing the behavior of the contextual and projection operators as well as the relations between these operators and the displacement one. First, the axiomatic system for *MLTL* must constrain $\prec$ to be a total order and $\subset$ to be a partial order that refines $\prec$, that is, for every pair of contexts c, c' we have that if $c \subset c'$, then $c \prec c'$, but not necessarily vice versa. Moreover, it must express the basic logical properties of the contextual and projection operators:

(AxNC)	$\nabla^c(\phi \rightarrow \psi) \rightarrow (\nabla^c\phi \rightarrow \nabla^c\psi)$	(normality of ∇^c);
(AxNP)	$\Box(\phi \rightarrow \psi) \rightarrow (\Box\phi \rightarrow \Box\psi)$	(normality of $\Box$);
(AxNEC)	$\Delta^c\phi \rightarrow \phi$	("necessity" for Δ^c);
(AxIC)	$\nabla^c\nabla^c\phi \equiv \nabla^c\phi$	(idempotency of ∇^c);
(AxCCD)	$\nabla^c\nabla_\alpha\phi \equiv \nabla_\alpha\nabla^c\phi$	(commutativity of ∇^c and ∇_α),

together with the rules:

(C-NEC)	$\vdash \phi \longrightarrow \vdash \nabla^c\phi$	(necessitation rule for ∇^c);
(P-NEC)	$\vdash \phi \longrightarrow \vdash \Box\phi$	(necessitation rule for $\Box$).

Notice that the projection operators $\Diamond$ and $\Box$ behave as the usual modal operators of possibility and necessity, while the contextual operators Δ^c and ∇^c are less standard (a number of theorems that account for the behavior of the contextual operators are given in [Montanari, 1996]). The set of axioms must also include the Barcan formula for the contextual and projection operators:

(AxBFC)	$\forall x\nabla^c\phi \rightarrow \nabla^c\forall x\phi$, with $x \neq c$	(Barcan formula for ∇^c);
(AxBFP)	$\forall x\Box\phi \rightarrow \Box\forall x\phi$	(Barcan formula for $\Box$),

as well as the counterparts of axioms (AxAD1)-(AxAD4) for the contextual operator. Similar axioms must be used to constrain the relationships between context terms, ordered by $\prec$ or $\subset$, and the displacement and contextual operators. Finally, we add a number of axioms that express the properties of the temporal structure, that is, the structural properties of the contextualization and projection relations. As an example, the axiom $\forall c_1, c_2, c_3((c_3 \subset c_2 \subset c_1 \wedge \nabla^{c_1}\Box\nabla^{c_3}\phi) \rightarrow \nabla^{c_1}\Box\nabla^{c_2}\Box\nabla^{c_3}\phi)$ can be added to constrain the projection relation to be downward transitive. Different classes of structures (e.g., homogeneous and non-homogeneous) can be captured by different sets of axioms. A sound axiomatic system for *MLTL* is reported in [Montanari *et al.*, 1992]. No completeness proof is given. In principle, one can try to directly prove it by building a canonical model for *MLTL*. However, even though there seem to be no specific technical problems to solve, the process of canonical model construction is undoubtedly very demanding in view of the size and complexity of the *MLTL* axiom system. As a matter of fact, one can follow an alternative approach, based on the technique proposed by Finger and Gabbay in [Finger and Gabbay, 1996], which views temporal logics for time granularity as combinations of simpler temporal logics, and specifies what constraints such combinations must satisfy to guarantee the transference of logical properties (including completeness results) from the component logics to the combined ones. In Section 3.4.3 we shall present temporal logics for time granularity which are obtained as suitable combinations of existing linear and branching temporal logics.

We conclude the section with a discussion of two classical problems about granularity conversions. The first problem has already been pointed out at the beginning of the section: given the truth value of a formula with respect to a certain layer, can we constrain (and how) its truth value with respect to the other layers? In [Montanari and Policriti, 1996], Montanari and Policriti give an example of a proposition which is true at every point of a given layer, and false with respect to every point of another one. It follows that, in general, we can record the links explicitly provided by the user, but we cannot impose any other constraint about the truth value of a formula with respect to a layer different from the layer it is associated with. Accordingly, *MLTL* makes it possible to write formulas involving granularity changes, but the proposed axiomatic systems do not impose any general constraint on the relations among the truth values of a formula with respect to different layers. Nevertheless, from a practical point of view, it makes sense to look for general rules expressing *typical* relations among truth values. In [Ciapessoni *et al.*, 1993], Ciapessoni et al. introduce two consistency rules that respectively allow one to project simple *MLTL* formulas, that is, *MLTL* formulas devoid of any occurrence of the displacement, contextual, and projection operators, from coarser to finer layers (downward temporal projection) and from finer to coarser ones (upward temporal projection). For any given pair of layers T^i, T^j, with $T^j \prec T^i$, any point $x \in T^i$, and any simple formula ϕ, *downward temporal projection* states that if ϕ holds at x, then there exists at least one $y \in T^j$ such that $\updownarrow (x, y)$ and ϕ holds at y, while *upward temporal projection* states that if ϕ holds at every $y \in T^j$ such that $\updownarrow (x, y)$, then ϕ holds at x. Formally, downward temporal projection is defined by the formula $\forall c_1, c_2(c_2 \subset c_1 \rightarrow \nabla^{c_1}(\phi \rightarrow \Diamond\Delta^{c_2}\phi))$, while upward temporal projection is defined by the formula $\forall c_1, c_2(c_2 \subset c_1 \rightarrow \nabla^{c_1}(\Box\nabla^{c_2}\phi \rightarrow \phi))$. It is not difficult to show that the two formulas are *inter-deducible* [Montanari, 1996]. (Downward) temporal projection captures the *weakest semantics* that can be attached to a statement with respect to a layer finer than the original one, provided that the statement is not wholistic. In most cases, however, such semantics is too weak, and additional user qualifications are needed. Various domain-specific categorizations of statements have been proposed in the literature [Roman, 1990; Shoham, 1988], which allow one to classify statements according to their behavior under temporal projection, e.g., events, properties, facts, and processes. In [Montanari, 1994], Montanari proposes some specializations of the *MLTL* projection operator $\Diamond$ that allow one to define different types of temporal projection, distinguishing among statements that hold at one and only one point of the decomposition interval (*punctual*), statements that hold at every point of such an interval (*continuous and pervasive*), statements that hold over a scattered sequence of sub-intervals of the decomposition interval (*bounded sequence*), and so on.

The second problem is the synchronization problem. We introduced this problem in Section 3.2, where we showed that the interpretations of the statements "tomorrow I will eat" and "dinner will be ready in one hour" with respect to a layer finer than the layer they explicitly refer to differ a lot. It is not difficult to show that even the same statement may admit different interpretations with respect to different finer layers (a detailed example can be found in [Montanari, 1996]). In general, the synchronization problem arises when logical formulas which state that a given fact holds at a point y of a layer T^i at distance α from the current point x need to be interpreted with respect to a finer layer T^j. There exist at least two possible interpretations for the original formula with respect to T^j (for the sake of simplicity, we restrict our attention to facts encoded by simple *MLTL* formulas, with a punctual interpretation under temporal projection, and we assume the temporal universe to be homogeneous). The first interpretation maps x (resp. y) into an arbitrary point x' (resp. y') of its decomposition interval, thus allowing the distance α' between x' and y' to vary. If x precedes y, we get the minimum (resp. maximum) value for α' when x' is the last (resp. first) element of the decomposition interval for x and y' is the first (resp. last) element of the decomposition interval for y. The second interpretation forces the mapping for y to conform to the mapping for x. As an example, if x is mapped into the first element of its decomposition interval, then y is mapped into the first element of its decomposition interval as well. As a consequence, there exists only one possible value for the distance α'. The first interpretation can be easily expressed in *MLTL* (it is the interpretation underlying the semantics of the projection operator). In order to enable *MLTL* to support the second interpretation, two extensions are needed: (i) we must replace the notion of current point by the notion of *vector* of current points (one for each layer); (ii) we must define a new projection operator that maps the current point of T^i into the current point of T^j, for every pair of layers T^i, T^j. Such extensions are accomplished in [Montanari, 1994]. In particular, it is possible to show that the new projection operator is second-order definable in terms of the original one, and that both projection operators are (second-order) definable in terms of a third simpler projection operator that maps every point into the first elements of its decomposition (and abstraction) intervals.

3.4.2 Monadic theories of time granularity

We move now from the temporal logic setting to the classical one, focusing our attention on monadic theories of time granularity. First, we introduce the relational structures for time granularity; then we present the theories of such structures and we analyze their decision problem. At the end, we briefly study the definability and decidability of meaningful binary predicates for time granularity with respect to such theories and some fragments of them.

Relational structures for time granularity

We begin with some preliminary definitions about finite and infinite sequences and trees (we assume the reader to be familiar with the notation and the basic notions of the theory of formal languages). Let A be a finite set of symbols and A^* be its Kleene closure. The length of a string $x \in A^*$, denoted by $|x|$, is defined in the usual way: $|\epsilon|$=0, $|xa| = |x| + 1$. For any pair $x, y \in A^*$, we say that x is a *prefix* of y, denoted by $x <_{pre} y$, if $xw = y$ for some $w \in A^+ (= A^* \setminus \{\epsilon\})$. The *prefix* relation $<_{pre}$ is a partial ordering over A^*. If A is totally ordered, a total ordering over A^* can be obtained from the one over A as follows. Let $<$ be

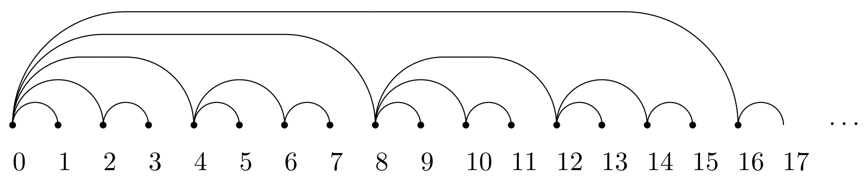

Figure 3.1: The structure of the relation $\mathtt{flip}_2$.

the total ordering over A. For every $x, y \in A^*$, we say that x lexicographically precedes y with respect to $<$, denoted $x <_{lex} y$, if either $x <_{pre} y$ or there exist $z \in A^*$ and $a, b \in A$ such that $za \leq_{pre} x$, $zb \leq_{pre} y$, and $a < b$. The *lexicographical* relation $<_{lex}$ is a total ordering over A^*.

A *finite sequence* is a relational structure $s = \langle I, < \rangle$, where I is an initial segment of the natural numbers $\mathbb{N}$ and $<$ is the usual ordering over $\mathbb{N}$. Given a *finite* set of monadic predicate symbols $\mathcal{P}$, a $\mathcal{P}$-labeled finite sequence is a relational structure $s_{\mathcal{P}} = \langle s, (\overline{P})_{P \in \mathcal{P}} \rangle$, where $s = \langle I, < \rangle$ and, for every $P \in \mathcal{P}$, $\overline{P} \subseteq I$ is the set of elements labeled with P (note that $\overline{P} \cap \overline{Q}$, with $P, Q \in \mathcal{P}$, can obviously be nonempty). An *infinite sequence* (ω-sequence for short) is a relational structure $s = \langle \mathbb{N}, < \rangle$ and a $\mathcal{P}$-labeled ω-sequence $s_{\mathcal{P}}$ is an ω-sequence s expanded with the sets $\overline{P}$, for $P \in \mathcal{P}$. For the sake of simplicity, hereafter we shall use the symbol P to denote both a monadic predicate and its interpretation; accordingly, we shall rewrite $s_{\mathcal{P}}$ as $\langle s, (P)_{P \in \mathcal{P}} \rangle$. In the following, we shall take into consideration three binary relations over $\mathbb{N}$, namely, $\mathtt{flip}_k$, $\mathtt{adj}$, and $2\times$. Let $k \geq 2$. The binary relation $\mathtt{flip}_k$ is defined as follows. Given $x, y \in \mathbb{N}$, $\mathtt{flip}_k(x, y)$, also denoted $\mathtt{flip}_k(x) = y$, if $y = x - z$, where z is the least power of k with non-null coefficient in the k-ary representation of x. Formally, $\mathtt{flip}_k(x) = y$ if $x = a_n \cdot k^n + a_{n-1} \cdot k^{n-1} + \ldots + a_m \cdot k^m$, $0 \leq a_i \leq k - 1$, $a_m \neq 0$, and $y = a_n \cdot k^n + a_{n-1} \cdot k^{n-1} + \ldots + (a_m - 1) \cdot k^m$. For instance, $\mathtt{flip}_2(18, 16)$, since $18 = 1 \cdot 2^4 + 1 \cdot 2^1$, $m = 1$, and $16 = 1 \cdot 2^4 + 0 \cdot 2^1$, while $\mathtt{flip}_2(16, 0)$, since $16 = 1 \cdot 2^4$, $m = 4$, and $0 = 0 \cdot 2^4$. Note that there exists no y such that $\mathtt{flip}_2(0, y)$. The structure of $\mathtt{flip}_2$ is depicted in Figure 3.1. The relation $\mathtt{adj}$ is defined as follows: $\mathtt{adj}(x, y)$, also denoted $\mathtt{adj}(x) = y$, if $x = 2^{k_n} + 2^{k_{n-1}} + \ldots + 2^{k_0}$, with $k_n > k_{n-1} > \ldots > k_0 > 0$, and $y = x + 2^{k_0} + 2^{k_0 - 1}$. For instance, $\mathtt{adj}(12, 18)$, since $12 = 2^3 + 2^2$, $k_0 = 2$, and $18 = 12 + 2^2 + 2^1$, while there exists no y such that $\mathtt{adj}(13, y)$, since $13 = 2^3 + 2^2 + 2^0$ and $k_0 = 0$. Finally, for any pair $x, y \in \mathbb{N}$, it holds that $2 \times (x, y)$ if $y = 2x$.

Finite and infinite (k-ary) trees are defined as follows. Let $k \geq 2$ and T_k be the set $\{0, \ldots k - 1\}^*$. A set $D \subseteq T_k$ is a (k-ary) *tree domain* if:

1. D is *prefix closed*, that is, for every $x, y \in T_k$, if $x \in D$ and $y <_{pre} x$, then $y \in D$;

2. for every $x \in T_k$, either $xi \in D$ for every $0 \leq i \leq k - 1$ or $xi \notin D$ for every $0 \leq i \leq k - 1$.

Note that, according to the definition, the whole T_k is a tree domain. A *finite tree* is a relational structure $\kappa = \langle D, (\downarrow_i)_{i=0}^{k-1}, <_{pre} \rangle$, where D is a finite tree domain, for every $0 \leq i \leq k-1$, $\downarrow_i$ is the i-th *successor relation* over D such that $\downarrow_i (x, y)$, also denoted $\downarrow_i (x) = y$, if $y = xi$, and $<_{pre}$ is the prefix ordering over D defined as above. The elements of D are

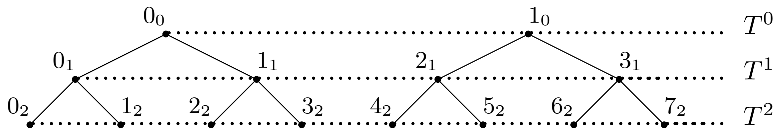

Figure 3.2: The 2-refinable 3-layered structure.

called *nodes*. If $\downarrow_i (x) = y$, then y is said the i-th *son* of x. The lexicographical ordering $<_{lex}$ over D is defined with respect to the natural ordering $<$ over $\{0, \ldots k-1\}$ such that $0 < 1 < \ldots < k-1$. A *path* P in κ is a subset of D whose nodes can be written as a sequence $x_0, x_1, \ldots$ such that, for every $i > 0$, there exists $0 \leq j \leq k-1$ with $x_i = \downarrow_j (x_{i-1})$. We shall denote by $P(i)$ the i-th element x_i of the path P. A *full path* is a maximal path with respect to set inclusion. A *chain* is any subset of a path. The *root* of κ is the node ϵ. A *leaf* of κ is an element $x \in D$ devoid of sons. A node which is not a leaf is called an *internal node*. The *depth* of a node $x \in D$ is the length of the (unique) path from the root ϵ to x. The *height* of κ is the maximum of the depths of the nodes in D. κ is *complete* if every leaf has the same depth. A $\mathcal{P}$-labeled finite tree is a relational structure $\kappa = \langle D, (\downarrow_i)_{i=0}^{k-1}, <_{pre}, (P)_{P \in \mathcal{P}} \rangle$, where the tuple $(D, (\downarrow_i)_{i=0}^{k-1}, <_{pre})$ is a finite tree and, for every $P \in \mathcal{P}$, $P \subseteq D$ is the set of nodes labeled with P. As for infinite trees, we are interested in *complete* infinite trees over the tree domain T_k. The complete *infinite tree* over T_k is the tuple $\kappa = \langle T_k, (\downarrow_i)_{i=0}^{k-1}, <_{pre} \rangle$. Paths, full paths, and chains are defined as for finite trees. A $\mathcal{P}$-labeled infinite tree is an expansion of the complete infinite tree over T_k with monadic predicates P, for $P \in \mathcal{P}$.

Relational structures for time granularity consists of a (possibly infinite) number of distinct layers/domains (we shall use the two terms interchangeably). We focus our attention on n-layered structures, which include a fixed finite number n of layers, and ω-layered structures, which feature an infinite number of layers.

Let $n \geq 1$ and $k \geq 2$. For every $0 \leq i < n$, let $T^i = \{j_i \mid j \geq 0\}$. The *$n$-layered temporal universe* is the set $\mathcal{U}_n = \bigcup_{0 \leq i < n} T^i$. The ($k$-refinable) *$n$-layered structure* ($n$-LS for short) is the relational structure $\langle \mathcal{U}_n, (\downarrow_j)_{j=0}^{k-1}, < \rangle$. Such a structure can be viewed as an infinite sequence of complete (k-ary) trees of height $n-1$, each one rooted at a point of the coarsest layer T^0 (see Figure 3.2). The sets T^i, with $0 \leq i < n$, are the layers of the trees. For every $0 \leq j \leq k-1$, $\downarrow_j$ is the j-th *successor relation* over $\mathcal{U}_n$ such that $\downarrow_j (x, y)$ (also denoted $\downarrow_j (x) = y$) if y is the j-th son of x. Hereafter, to adhere to the common terminology in the field, we shall substitute the term projection for the term successor. Note that for all x belonging to the finest layer T^{n-1} there exist no $0 \leq j \leq k-1$ and $y \in \mathcal{U}_n$ such that $\downarrow_j (x) = y$. Finally, $<$ is a total ordering over $\mathcal{U}_n$ given by the *pre-order* (root-left-right in the binary trees) visit of the nodes (for elements belonging to the same tree) and by the total linear ordering of trees (for elements belonging to different trees). Formally, for any pair $a_b, c_d \in \mathcal{U}_n$, we have that $\downarrow_j (a_b) = c_d$ if $b < n-1$, $d = b+1$, and $c = a \cdot k + j$. The total ordering $<$ is defined as follows:

1. if $x = a_0$, $y = b_0$, and $a < b$ over $\mathbb{N}$, then $x < y$;

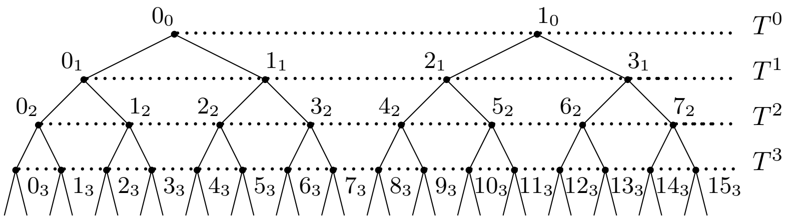

Figure 3.3: The 2-refinable downward unbounded layered structure.

2. for all $x \in \mathcal{U}_n \setminus T^{n-1}$, $x <\downarrow_0 (x)$, and $\downarrow_j (x) <\downarrow_{j+1} (x)$, for all $0 \leq j < k-1$;
3. if $x \in \mathcal{U}_n \setminus T^{n-1}$, $x < y$, and not $\texttt{ancestor}(x, y)$, then $\downarrow_{k-1} (x) < y$;
4. if $x < z$ and $z < y$, then $x < y$,

where $\texttt{ancestor}(x, y)$ if there exists $0 \leq j \leq k-1$ such that $\downarrow_j (x) = y$ or there exist $0 \leq j \leq k-1$ and z such that $\downarrow_j (z) = y$ and $\texttt{ancestor}(x, z)$. A *path* over the n-LS is a subset of the domain whose elements can be written as a sequence $x_0, x_1, \ldots x_m$, with $m \leq n-1$, in such a way that, for every $i = 1, \ldots m$, there exists $0 \leq j < k$ for which $x_i =\downarrow_j (x_{i-1})$. A *full path* is a maximal path with respect to set inclusion. A *chain* is any subset of a path. A $\mathcal{P}$-labeled n-LS is a relational structure $\langle \mathcal{U}_n, (\downarrow_i)_{i=0}^{k-1}, <, (P)_{P \in \mathcal{P}} \rangle$, where the tuple $(\mathcal{U}_n, (\downarrow_i)_{i=0}^{k-1}, <)$ is the n-LS and, for every $P \in \mathcal{P}$, $P \subseteq \mathcal{U}_n$ is the set of points labeled with P.

As for ω-layered structures, we focus our attention on the (k-refinable) downward unbounded layered structure (DULS for short), which consists of a coarsest domain together with an infinite number of finer and finer domains, and the (k-refinable) upward unbounded layered structure (UULS for short), which consists of a finest temporal domain together with an infinite number of coarser and coarser domains. Let $\mathcal{U} = \bigcup_{i \geq 0} T^i$ be the *ω-layered temporal universe*. The DULS is a relational structure $\langle \mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, < \rangle$. It can be viewed as an infinite sequence of complete (k-ary) infinite trees, each one rooted at a point of the coarsest domain T^0 (see Figure 3.3). The sets T^i, with $i \geq 0$, are the layers of the trees. The definitions of the projection relations $\downarrow_j$, with $0 \leq j \leq k-1$, and the total ordering $<$ over $\mathcal{U}$ are close to those for the n-LS. Formally, for any pair $a_b, c_d \in \mathcal{U}$, we have that $\downarrow_j (a_b) = c_d$ if and only if $d = b+1$ and $c = a \cdot k + j$, while the total ordering $<$ is defined as follows:

1. if $x = a_0$, $y = b_0$, and $a < b$ over $\mathbb{N}$, then $x < y$;
2. for all $x \in \mathcal{U}$, $x <\downarrow_0 (x)$, and $\downarrow_j (x) <\downarrow_{j+1} (x)$, for all $0 \leq j < k-1$;
3. if $x < y$ and not $\texttt{ancestor}(x, y)$, then $\downarrow_{k-1} (x) < y$;
4. if $x < z$ and $z < y$, then $x < y$.

A *path* over the DULS is a subset of the domain whose elements can be written as an infinite sequence $x_0, x_1, \ldots$ such that, for every $i \geq 1$, there exists $0 \leq j < k$ for which $x_i =\downarrow_j (x_{i-1})$. A *full path* is a maximal (infinite) path with respect to set inclusion. A *chain* is

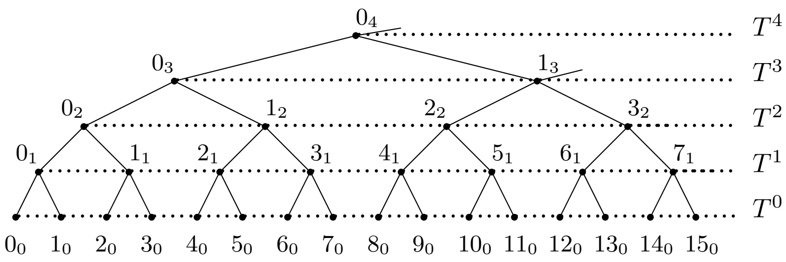

Figure 3.4: The 2-refinable upward unbounded layered structure.

any subset of a path. A $\mathcal{P}$-labeled DULS is a relational structure $\langle \mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, <, (P)_{P\in\mathcal{P}} \rangle$, where the tuple $(\mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, <)$ is the DULS and, for every $P \in \mathcal{P}$, $P \subseteq \mathcal{U}$ is the set of points labeled with P.

The UULS is a relational structure $\langle \mathcal{U}, \langle \downarrow_i)_{i=0}^{k-1}, < \rangle$. It can be viewed as a complete (k-ary) infinite tree generated from the leaves (Figure 3.4). The sets T^i, with $i \geq 0$, are the layers of the tree. For every $0 \leq j \leq k-1$, $\downarrow_j$ is the j-th *projection relation* over $\mathcal{U}$ such that $\downarrow_j (x, y)$ (also denoted by $\downarrow_j (x) = y$) if y is the j-th son of x. The total ordering $<$ over $\mathcal{U}$ is induced by the *in-order* (left-root-right in the binary tree) visit of the treelike structure. Formally, for every $a_b, c_d \in \mathcal{U}$, $\downarrow_j (a_b) = c_d$ if $b > 0$, $d = b-1$, and $c = a \cdot k + j$. The total ordering $<$ is defined as follows:

1. for all $x \in \mathcal{U} \setminus T^0$, $\downarrow_0 (x) < x$, $x <\downarrow_1 (x)$, and $\downarrow_j (x) <\downarrow_{j+1} (x)$, for every $0 < j < k-1$;
2. if $x < y$ and not $\texttt{ancestor}(x, y)$, then $\downarrow_{k-1} (x) < y$;
3. if $x < y$ and not $\texttt{ancestor}(y, x)$, then $x <\downarrow_0 (y)$;
4. if $x < z$ and $z < y$, then $x < y$.

A *path* over the UULS is a subset of the domain whose elements can be written as an infinite sequence $x_0, x_1, \ldots$ such that, for every $i \geq 1$, there exists $0 \leq j < k$ such that $x_{i-1} =\downarrow_j (x_i)$. A *full path* is a maximal (infinite) path with respect to set inclusion. A *chain* is any subset of a path. It is worth noting that every pair of paths over the UULS may differ on a finite prefix only. A $\mathcal{P}$-labeled UULS is obtained by expanding the UULS with a set $P \subseteq \mathcal{U}$, for any $P \in \mathcal{P}$.

Theories of time granularity

We are now ready to introduce the theories of time granularity. They are systems of monadic second-order (MSO for short) logic that allow quantification over arbitrary sets of elements. We shall study the properties of the full systems as well as of some meaningful fragments of them. We shall show that some granularity theories can be reduced to well-know classical MSO theories, such as the MSO theory of one successor and the MSO theory of two successors, while other granularity theories are proper extensions of them.

Definition 3.4.1. (*The language of monadic second-order logic*)

Let $\tau = c_1, \ldots, c_r, u_1, \ldots, u_s, b_1, \ldots, b_t$ be a finite alphabet of symbols, where $c_1, \ldots, c_r$ (resp. $u_1, \ldots, u_s$, $b_1, \ldots, b_t$) are constant symbols (resp. unary relational symbols, binary relational symbols), and let $\mathcal{P}$ be a finite set of uninterpreted unary relational symbols. The second-order language with equality $\mathrm{MSO}[\tau \cup \mathcal{P}]$ is built up as follows:

1. atomic formulas *are of the forms $x = y$, $x = c_i$, with $1 \leq i \leq r$, $u_i(x)$, with $1 \leq i \leq s$, $b_i(x, y)$, with $1 \leq i \leq t$, $x \in X$, $x \in P$, where x, y are individual variables, X is a set variable, and $P \in \mathcal{P}$;*
2. formulas *are built up from atomic formulas by means of the Boolean connectives $\neg$ and $\wedge$, and the quantifier $\exists$ ranging over both individual and set variables.*

In the following, we shall write $\mathrm{MSO}_{\mathcal{P}}[\tau]$ for $\mathrm{MSO}[\tau \cup \mathcal{P}]$; in particular, we shall write $\mathrm{MSO}[\tau]$ when $\mathcal{P}$ is meant to be the empty set. The first-order fragment of $\mathrm{MSO}_{\mathcal{P}}[\tau]$ will be denoted by $\mathrm{FO}_{\mathcal{P}}[\tau]$, while its path (resp. chain) fragment, which is obtained by interpreting second-order variables over paths (resp. chains), will be denoted by $\mathrm{MPL}_{\mathcal{P}}[\tau]$ (resp. $\mathrm{MCL}_{\mathcal{P}}[\tau]$). We focus our attention on the following theories:

1. $\mathrm{MSO}_{\mathcal{P}}[<]$ and its first-order fragment interpreted over finite and ω-sequences;
2. $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ (as well as its first-order fragment), $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$, and $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ interpreted over ω-sequences;
3. $\mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$ and its first-order, path, and chain fragments interpreted over finite and infinite trees;
4. $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ and its first-order, path, and chain fragments interpreted over the n-LS, the DULS, and the UULS.

We preliminarily introduce some notations and basic properties that will help us in comparing the expressive power and logical properties of the various theories. Most definitions and results are given for full MSO theories with uninterpreted unary relational symbols, but they immediately transfer to their fragments, possibly devoid of uninterpreted unary relational symbols.
Let $\mathcal{M}(\varphi)$ be the set of models of the formula φ. We say that $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ can be *embedded* into $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$, denoted $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_2]$, if there is an *effective* translation tr of $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$-formulas into $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$-formulas such that, for every formula $\varphi \in \mathrm{MSO}_{\mathcal{P}}[\tau_1]$, $\mathcal{M}(\varphi) = \mathcal{M}(tr(\varphi))$. For instance, it is easy to prove that $\mathrm{FO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}] \rightarrow \mathrm{MPL}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}] \rightarrow \mathrm{MCL}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}] \rightarrow \mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$ (the same holds for their counterparts devoid of $\mathcal{P}$), where all theories are interpreted over trees. The condition 'X is a path' can indeed be written in the monadic chain logic, and the condition 'X is a chain' can be expressed in the MSO logic. It is also easy to show that the monadic path logic over paths is as expressive as the monadic path logic over full paths. Moreover, we say that $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ is *as expressive as* $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$, written $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightleftarrows \mathrm{MSO}_{\mathcal{P}}[\tau_2]$, if both $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_2]$ and $\mathrm{MSO}_{\mathcal{P}}[\tau_2] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_1]$. It is immediate to see that if $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_2]$ and $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$ is decidable (resp. $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ is undecidable), then $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ is decidable (resp. $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$ is undecidable) as well.
Besides decidability issues, we are interested in definability ones. Let β be a relational symbol. We say that β is *definable* in $\mathrm{MSO}_{\mathcal{P}}[\tau]$ if $\mathrm{MSO}_{\mathcal{P}}[\tau \cup \{\beta\}] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau]$. If the

addition of β to a decidable theory $\mathrm{MSO}_{\mathcal{P}}[\tau]$ makes the resulting theory $\mathrm{MSO}_{\mathcal{P}}[\tau \cup \{\beta\}]$ undecidable, we can conclude that β is not definable in $\mathrm{MSO}_{\mathcal{P}}[\tau]$. The opposite does not hold in general: the predicate β may not be definable in $\mathrm{MSO}_{\mathcal{P}}[\tau]$, but the extension of $\mathrm{MSO}_{\mathcal{P}}[\tau]$ with β may preserve decidability. In such a case, we obviously cannot reduce the decidability of $\mathrm{MSO}_{\mathcal{P}}[\tau \cup \{\beta\}]$ to that of $\mathrm{MSO}_{\mathcal{P}}[\tau]$.

The decidability of $\mathrm{MSO}_{\mathcal{P}}[<]$ over finite sequences has been proved in [Büchi, 1960; Elgot, 1961], while its decidability over ω-sequences has been shown in [Büchi, 1962] ($\mathrm{MSO}_{\mathcal{P}}[<]$ over ω-sequences is the well-known MSO theory of one successor $S1S$).

Theorem 3.4.2. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<]$ *over sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<]$ *over finite (resp. infinite) sequences is non-elementarily decidable.*

The theory $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ ($S1S^k$ for short), interpreted over ω-sequences, has been studied by Monti and Peron in [Monti and Peron, 2000]. Such a theory properly extends $S1S$. Moreover, the unary predicate $\mathtt{pow}_k$ such that $\mathtt{pow}_k(x)$ if x is a power of k can be easily expressed as $\mathtt{flip}_k(x) = 0$. Hence, $S1S^k$ is at least as expressive as the well-known (decidable) extension of $\mathrm{MSO}_{\mathcal{P}}[<]$ with the predicate $\mathtt{pow}_k$ [Elgot and Rabin, 1966]. The decidability of $S1S^k$ has been proved by showing that it is the logical counterpart of the class of ω-sequences languages (ω-languages for short) recognized by systolic (k-ary) tree automata. The class of the languages of finite sequences recognized by systolic tree automata was originally investigated by Culik II et al. in [Culik II *et al.*, 1984]. In [Monti and Peron, 2000], Monti and Peron extend the notion of systolic tree automaton to deal with ω-languages. They prove that the class of systolic tree ω-languages is a proper extension of the class of regular ω-languages (that is, ω-languages recognized by Büchi automata), that maintains the closure properties of regular ω-languages as well as the decidability of the emptiness problem. The correspondence between systolic tree ω-languages and $S1S^k$ is established by means of a generalization of Büchi's Theorem.

Theorem 3.4.3. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ *over* ω*-sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ *over* ω*-sequences is non-elementarily decidable.*

The theories $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ and $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$, interpreted over ω-sequences, have been investigated in [Monti and Peron, 2001]. $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ is a proper extension $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_2]$. Unfortunately, unlike $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_2]$, it is undecidable.

Theorem 3.4.4. (*Undecidability of* $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ *over* ω*-sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ *over infinite sequences is undecidable.*

Since $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ is at least as expressive as $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$, its decision problem is undecidable as well.

Theorem 3.4.5. (*Undecidability of* $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ *over* ω*-sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ *over* ω*-sequences is undecidable.*

The theories $\mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$, interpreted over infinite (k-ary) trees, are the well-known MSO theories of k successors (SkS for short). The decidability of SkS over finite trees has been shown in [Doner, 1970; Thatcher and Wright, 1968]. The decidability of the MSO theory of the infinite binary tree $S2S$ has been proved in [Rabin, 1969]. Such a result can be easily generalized to the MSO theory of the infinite k-ary tree SkS, for any $k > 2$ (and even to $S\omega S$ over countably branching trees) [Thomas, 1990].

Theorem 3.4.6. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$ *over trees*)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over finite (resp. infinite) trees is non-elementarily decidable.*

The decidability of $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ over the n-LS has been proved in [Montanari and Policriti, 1996] by reducing it to $S1S$. Such a reduction is accomplished in two steps. First, the n-layered structure is flattened by embedding all its layers into the finest one; then, metric temporal information is encoded by means of a finite set of unary relations. This second step is closely related to the technique exploited in [Alur and Henzinger, 1993] to prove the decidability of a family of real-time logics*. It relies on the *finite-state character* of the involved metric temporal information, which can be expressed as follows: every temporal property that partition an infinite set of states/time points into a finite set of classes can be finitely *modeled* and hence it is decidable.

Theorem 3.4.7. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* n-LS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* n*-LS is non-elementarily decidable.*

The decidability of $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ over both the DULS and the UULS has been shown in [Montanari *et al.*, 1999]. The decidability of the theory of the DULS has been proved by embedding it into SkS. The infinite sequence of infinite trees of the k-refinable DULS can indeed be appended to the rightmost full path of the infinite k-ary tree. The encoding of the 2-refinable DULS into the infinite binary tree is shown in Figure 3.5. Suitable definable predicates are then used to distinguish between the nodes of the infinite tree that correspond to elements of the original DULS, and the other nodes. As an example, in the case depicted in Figure 3.5 we must differentiate the auxiliary nodes belonging to the rightmost full path of the tree from the other ones. Finally, for $0 \leq j \leq k-1$, the j-th projection relation $\downarrow_j$ can be interpreted as the j-th successor relation and the total order $<$ can be naturally mapped into the lexicographical ordering $<_{lex}$ (it is not difficult to show that $<_{lex}$ can be defined in SkS).

Theorem 3.4.8. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* DULS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the DULS is non-elementarily decidable.*

The decidability of the theory of the UULS has been proved by reducing it to $S1S^k$. For the sake of simplicity, we describe the basic steps of this reduction in the case of the 2-refinable UULS (the technique can be generalized to deal with any $k > 2$). An embedding of $MSO[<, \downarrow_0, \downarrow_1]$ into $S1S^2$ can be obtained as follows. First, we replace the 2-refinable ULLS by the so-called *concrete* 2-refinable ULLS, which is defined as follows:

- for all $i \geq 0$, the i-th layer T^i is the set $\{2^i + n2^{i+1} : n \geq 0\} \subseteq \mathbb{N}$;

*The relationships between the theories of n- and ω-layered structures and real-time logics have been explored in detail by Montanari et al. in [Montanari *et al.*, 2000]. Logic and computer science communities have traditionally followed a different approach to the problem of representing and reasoning about time and states. Research in logic resulted in a family of (metric) tense logics that take *time* as a primitive notion and define (*timed*) *states* as sets of atomic propositions which are true at given time points, while research in computer science concentrated on the so–called (real-time) temporal logics of programs that take *state* as a primitive notion, and define *time* as an attribute of states. Montanari et al. show that the theories of time granularity provide a unifying framework within which the two approaches can be reconciled. States and time-points can indeed be uniformly referred to as elements of the (decidable) theories of the DULS and the UULS. In particular, they show that the theory of timed state sequences, underlying real-time logics, can be naturally recovered as an abstraction of such theories.

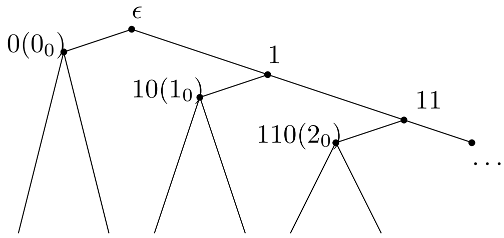

Figure 3.5: The encoding of the 2-refinable DULS into $\{0, 1\}^\star$.

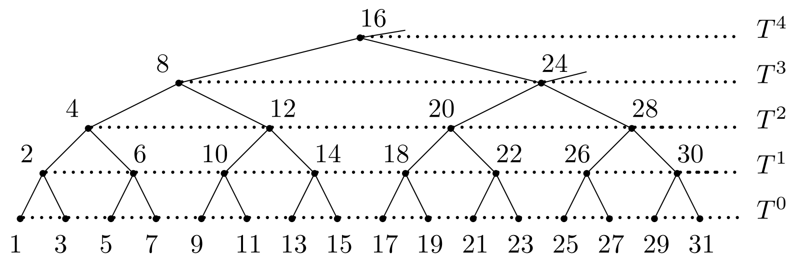

Figure 3.6: The concrete 2-refinable UULS.

- for every element $x = 2^i + n2^{i+1}$ belonging to T^i, with $i \geq 1$, $\downarrow_0 (x) = 2^i + n2^{i+1} - 2^{i-1} = 2^{i-1} + 2n2^i$ and $\downarrow_1 (x) = 2^i + n2^{i+1} + 2^{i-1} = 2^{i-1} + (2n+1)2^i$;
- $<$ is the usual ordering over $\mathbb{N}$.

A fragment of this concrete structure is depicted in Figure 3.6. Notice that all odd numbers are associated with layer T^0, while even numbers are distributed over the remaining layers. Notice also that the labeling of the concrete structure does not include the number 0^*. It is easy to show that the two structures are isomorphic by exploiting the obvious mapping that associates each element of the 2-refinable UULS with the corresponding element of the concrete structure, preserving projection and ordering relations. Hence, the two structures satisfy the same $MSO[<, \downarrow_0, \downarrow_1]$-formulas. Next, we can easily encode the concrete 2-refinable UULS into $\mathbb{N}$. Both relations $\downarrow_0$ and $\downarrow_1$ can indeed be defined in terms of $\texttt{flip}_2$ as follows. For any given even number x,

$$\begin{array}{lll} \downarrow_0 (x) = y & \text{iff} & y < x \wedge \texttt{flip}_2(y) = \texttt{flip}_2(x) \wedge \\ & & \neg \exists z (y < z \wedge z < x \wedge \texttt{flip}_2(z) = \texttt{flip}_2(x)); \\ \downarrow_1 (x) = y & \text{iff} & \texttt{flip}_2(y) = x \wedge \neg \exists z (y < z \wedge \texttt{flip}_2(z) = x). \end{array}$$

By exploiting such a correspondence, it is possible to define a translation τ of $MSO[<, \downarrow_0, \downarrow_1]$ formulas (resp. sentences) into $S1S^2$ formulas (resp. sentences) such that, for any formula

*In [Montanari *et al.*, 2002a], Montanari et al. show that it is convenient to consider 0 as the label of the first node of an imaginary additional finest layer, whose remaining nodes are not labeled. In such a way the node with label 0 turns out to be the left son of the node with label 1.

(resp. sentence) $\phi \in MSO[<, \downarrow_0, \downarrow_1]$, ϕ is satisfiable by (resp. true in) the UULS if and only if $\tau(\phi) \in S1S^2$ is satisfiable by (resp. true in) $\langle \mathbb{N}, <, \mathtt{flip}_2 \rangle$.

Theorem 3.4.9. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* UULS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the UULS is non-elementarily decidable.*

In [Montanari and Puppis, 2004b], Montanari and Puppis deal with the decision problem for the MSO logic interpreted over an ω-layered temporal structure devoid of both a finest layer and a coarsest one (we call such a structure totally unbounded, TULS for short). The temporal universe of the TULS is the set $\mathcal{U}_n = \bigcup_{i \in \mathbb{Z}} T^i$, where $\mathbb{Z}$ is the set of integers; the layer T^0 is a distinguished intermediate layer of such a structure. It is not difficult to show that $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ over both the DULS and the UULS can be embedded into $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ over the TULS (L_0 is a unary relational symbol used to identify the elements of T^0). The solution to the decision problem for $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ proposed by Montanari and Puppis extends Carton and Thomas' solution to the decision problem for the MSO theories of residually ultimately periodic words [Carton and Thomas, 2002]. First, they provide a tree-like characterization of the TULS and, taking advantage of it, they define a non-trivial encoding of the TULS into a vertex-colored tree that allows them to reduce the decision problem for the TULS to the problem of determining, for any given Rabin tree automaton, whether it accepts such a vertex-colored tree. Then, they reduce this latter problem to the decidable case of regular trees by exploiting a suitable notion of tree equivalence [Montanari and Puppis, 2004a].

Theorem 3.4.10. (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ *over the* TULS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ *over the TULS is non-elementarily decidable.*

Notice that, taking advantage of the above-mentioned embedding, such a result provides, as a by-product, an alternative (uniform) decidability proof for the theories of the DULS and the UULS.

The definability and decidability of a set of binary predicates in monadic languages interpreted over the n-LS, the DULS, and the UULS have been systematically explored in [Franceschet *et al.*, 2003]. The set of considered predicates includes the equi-level (resp. equi-column) predicate constraining two time points to belong to the same layer (resp. column) and the horizontal (resp. vertical) successor predicate relating a time point to its successor within a given layer (resp. column), which allow one to express meaningful properties of time granularity [Montanari, 1996]. The authors investigate definability and decidability issues for such predicates with respect to $\mathrm{MSO}[<, (\downarrow_i)_{i=0}^{k-1}]$ and its first-order, chain, and path fragments $\mathrm{FO}[<, (\downarrow_i)_{i=0}^{k-1}]$, $\mathrm{MPL}[<, (\downarrow_i)_{i=0}^{k-1}]$, and $\mathrm{MCL}[\tau]$ of $\mathrm{MSO}[<, (\downarrow_i)_{i=0}^{k-1}]$ (as well as their $\mathcal{P}$-variants $\mathrm{FO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$, $\mathrm{MPL}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$, and $\mathrm{MCL}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$). Figure 3.7 summarizes the relationships between the expressive powers of such formal systems (an arrow from $\mathcal{T}$ to $\mathcal{T}'$ stands for $\mathcal{T} \to \mathcal{T}'$). From Theorems 3.4.7, 3.4.8, 3.4.9, and 3.4.10, it immediately follows that all the formalisms in Figure 3.7, when interpreted over the n-LS, the DULS, the UULS, and the TULS are decidable.

The outcomes of the analysis of the equi-level, equi-column, horizontal successor, and vertical successor predicates can be summarized as follows. First, the authors show that all these predicates are not definable in the MSO language over the DULS and the UULS, and that their addition immediately leads the MSO theories of such structures to undecidability.

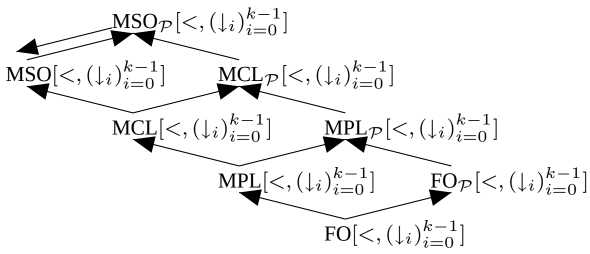

Figure 3.7: A hierarchy of monadic formalisms over layered structures.

As for the n-LS, the status of the horizontal (equi-level and horizontal successor) and vertical (equi-column and vertical successor) predicates turns out to be quite different: while horizontal predicates are easily definable, vertical ones are undefinable and their addition yields undecidability. Then, the authors study the effects of adding the above predicates to suitable *fragments* of the MSO language, such as its first-order, path, and chain fragments, possibly admitting uninterpreted unary relational symbols. They systematically explore all the possibilities, and give a number of positive and negative results. From a technical point of view, (un)definability and (un)decidability results are obtained by reduction from/to a wide spectrum of undecidable/decidable problems. Even though the complete picture is still missing (some decidability problems are open), the achieved results suffice to formulate some general statements. First, all predicates can be added to monadic first-order, path, and chain fragments, devoid of uninterpreted unary relational symbols, over the n-LS and the UULS preserving decidability. In the case of the DULS, they prove the same result for the equi-level and horizontal successor predicates, while they do not establish whether the same holds for the equi-column and vertical successor predicates. Moreover, they prove that the addition of the equi-column or vertical successor predicates to monadic first-order fragments over the ω-layered structures, with uninterpreted unary relational symbols, makes the resulting theories undecidable. The effect of such additions to the n-layered structure is not known. As for the equi-level predicate, they only prove that adding it to the monadic path fragment over the DULS, with uninterpreted unary relational symbols, leads to undecidability. Finally, as far as the MSO language over the UULS is concerned, they establish an interesting connection between its extension with the equi-level (resp. equi-column) predicate and systolic ω-languages over Y-trees (resp. trellis) [Gruska, 1990].

3.4.3 Temporalized logics and automata for time granularity

In the previous section, we have shown that monadic theories of time granularity are quite expressive, but they have not much computational appeal because their decision problem is *non-elementary*. This roughly means that it is possible to algorithmically check the truth of sentences, but the complexity of the algorithm grows very rapidly and it cannot be bounded. Moreover, the corresponding automata (Büchi sequence automata for the theory of the n-LS, Rabin tree automata for the theory of the DULS, and systolic tree automata for the theory of the UULS) do not directly work over layered structures, but rather over collapsed structures into which layered structures can be encoded. Hence, they are not natural and intuitive tools to specify and check properties of time granularity. In this section, we outline a different approach that connects monadic theories of time granularity back

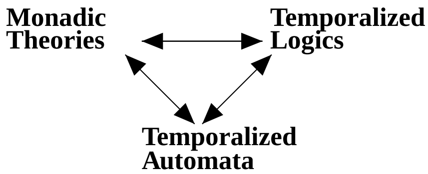

Figure 3.8: From monadic theories to temporalized logics via temporalized automata.

to temporal logic [Franceschet and Montanari, 2001a; Franceschet and Montanari, 2001b; Franceschet and Montanari, 2004]. Taking inspiration of methods for logic combinations (a short description of these methods can be found in [Franceschet *et al.*, 2004]), Franceschet and Montanari reinterpret layered structures as *combined structures*. This allows them to define suitable combined temporal logics and combined automata over layered structures, respectively called temporalized logics and temporalized automata, and to study their expressive power and computational properties by taking advantage of the transfer theorems for combined logics and combined automata. The outcome is rewarding: the resulting combined temporal logics and automata directly work over layered structures; moreover, they are expressively equivalent to monadic systems, and they are elementarily decidable.

Finding the temporal logic counterpart of monadic theories is a difficult task, involving a non-elementary blow up in the length of formulas. Ehrenfeucht games have been successfully exploited to deal with such a correspondence problem for first-order theories [Immerman and Kozen, 1989] and well-behaved fragments of second-order monadic ones, e.g., the path fragment of the monadic second-order theory of infinite binary trees [Hafer and Thomas, 1987]. As for the theories of time granularity, in [Franceschet and Montanari, 2003] Franceschet and Montanari show that an expressively complete and elementarily decidable combined temporal logic counterpart of the path fragment of the MSO theory of the DULS can be obtained by means of suitable applications of Ehrenfeucht games. Ehrenfeucht games have also been used by Montanari et al. to extend Kamp's theorem to deal with the first-order fragment of the MSO theory of the UULS [Montanari *et al.*, 2002a]. Unfortunately, these techniques produce rather involved proofs and they do not naturally lift to the full second-order case. A little detour is needed to deal with such a case. Instead of trying to establish a direct correspondence between MSO theories of time granularity and temporal logics, Franceschet and Montanari connect them via automata [Franceschet and Montanari, 2004] (cf. Figure 3.8). Firstly, they define the class of temporalized automata, which can be proved to be the automata-theoretic counterpart of temporalized logics, and they show that relevant properties, such as closure under Boolean operations, decidability, and expressive equivalence with respect to temporal logics, transfer from component automata to temporalized ones. Then, on the basis of the established correspondence between temporalized logics and automata, they reduce the task of finding a temporal logic counterpart of the MSO theories of the DULS and the UULS to the easier one of finding temporalized automata counterparts of them. The mapping of MSO formulas into automata (the difficult direction) can indeed greatly benefit from automata closure properties.

As a by-product, the alternative characterization of temporalized logics for time granularity as temporalized automata allows one to reduce logical problems to automata ones. As it is well-known in the area of automated system specification and verification, such a reduction presents several advantages, including the possibility of using automata for both system modeling and specification, and the possibility of checking the system on-the-fly (a detailed account of these advantages can be found in [Franceschet and Montanari, 2001b]).

3.4.4 Coda: time granularity and interval temporal logics

As pointed out in [Montanari, 1996], there exists a natural link between structures and theories of time granularity and those developed for representing and reasoning about time intervals. Differently-grained temporal domains can indeed be interpreted as different ways of partitioning a given discrete/dense time axis into consecutive disjoint intervals. According to this interpretation, every time point can be viewed as a suitable interval over the time axis and projection implements an intervals-subintervals mapping. More precisely, let us define *direct constituents* of a time point x, belonging to a given domain, the time points of the immediately finer domain into which x can be refined, if any, and *indirect constituents* the time points into which the direct constituents of x can be directly or indirectly refined, if any. The mapping of a given time point into its direct or indirect constituents can be viewed as a mapping of a given time interval into (a specific subset of) its subintervals.

The existence of such a natural correspondence between interval and granularity structures hints at the possibility of defining a similar connection at the level of the corresponding theories. For instance, according to such a connection, temporal logics over DULSs allow one to constrain a given property to hold true densely over a given time interval, where P densely holds over a time interval w if P holds over w and there exists a direct constituent of w over which P densely holds. In particular, establishing a connection between structures and logics for time granularity and those for time intervals would allow one to transfer decidability results from the granularity setting to the interval one. As a matter of fact, most interval temporal logics, including Moszkowski's Interval Temporal Logic (ITL) [Moszkowski, 1983], Halpern and Shoham's Modal Logic of Time Intervals (HS) [Halpern and Shoham, 1991], Venema's CDT Logic [Venema, 1991a], and Chaochen and Hansen's Neighborhood Logic (NL) [Chaochen and Hansen, 1998], are highly undecidable. Decidable fragments of these logics have been obtained by imposing severe restrictions on their expressive power, e.g., the *locality* constraint in [Moszkowski, 1983].

Preliminary results can be found in [Montanari *et al.*, 2002b], where the authors propose a new interval temporal logic, called Split Logic (SL for short), which is equipped with operators borrowed from HS and CDT, but is interpreted over specific interval structures, called *split-frames*. The distinctive feature of a split-frame is that there is at most one way to chop an interval into two adjacent subintervals, and consequently it does not possess *all* the intervals. They prove the decidability of SL with respect to particular classes of split-frames which can be put in correspondence with the first-order fragments of the monadic theories of time granularity. In particular, *discrete* split-frames with maximal intervals correspond to the n-layered structure, discrete split-frames (with unbounded intervals) can be mapped into the upward unbounded layered structure, and *dense* split-frames with maximal intervals can be encoded into the downward unbounded layered structure.

3.5 Qualitative time granularity

Granularity operators for qualitative time representation have been first provided in [Euzenat, 1993; Euzenat, 1995a]. These operators are defined in the context of relational algebras and they apply to both point and interval algebras. They have the advantage of being applicable to fully qualitative and widespread relational representations. They account for granularity phenomena occurring in actual applications using only qualitative descriptions.

After a short recall of relation algebras (Section 3.5.1), a set of six constraints applying to the granularity operators is defined (Section 3.5.2). These constraints are applied to the well-known temporal representation of point and interval algebras (Section 3.5.3). Some general results of existence and relation of these operators with composition are also given (Section 3.5.4).

3.5.1 Qualitative time representation and granularity

The qualitative time representation considered here is a well-known one:

1. it is based on an algebra of binary relations $\langle 2^{\Gamma}, \cup, \circ, ^{-1} \rangle$ (see Chapter 1); we focus our attention on the point and interval algebras [Vilain and Kautz, 1986; Allen, 1983]);
2. this algebra is augmented with a neighborhood structure (in which $N(r, r')$ means that the relationships r and r' are neighbors) [Freksa, 1992];
3. last, the construction of an interval algebra [Hirsh, 1996] is considered (the conversion of a quadruple of base relationships R into an interval relation is given by $\Rightarrow R$ and the converse operation by $\Leftarrow r$ when it is defined).

In such an algebra of relations, the situations are described by a set of possible relationships holding between entities (here points or intervals).

As an example, imagine several witnesses of an air flight incident with the witness from the ground (g) saying that "the engine stopped working (W) and the plane went [immediately] down", the pilot (p) saying that "the plane worked correctly (W) until there has been a misfiring period (M) and, after that, the plane lost altitude", and the (unfortunately out of reach) "blackbox" flight data recorder (b) revealing that the plane had a short misfiring period (M) and a short laps of correct behavior before the plane lost altitude (D).
If these descriptions are rephrased in the interval algebra (see Figure 3.9), this would correspond to three different descriptions: $g = \{WmD\}$, $p = \{WmM, MmD\}$ and $b = \{WmM, MbD\}$. Obviously, if any two of these descriptions are merged, the result is an inconsistent description. However, such inconsistencies arise because the various sources of information do not share the same precision and not because of intrisically contradictory descriptions. It is thus useful to find in which way the situations described by g and p can be coarse views of that expressed by b.

The qualitative granularity is defined through a couple of operators for converting the representation of a situation into a finer or coarser representation of the same situation. These operators apply to the relationships holding between the entities and transform these relationship into other plausible relationships at a coarser (with upward conversion denoted by $\uparrow$) or finer (with downward conversion denoted by $\downarrow$) granularity. When the conversion is not oriented, i.e., when we talk about a granularity change between two layers, but it is not necessary to know which one is the coarser, a neutral operator is used (denoted by $\rightarrow$).

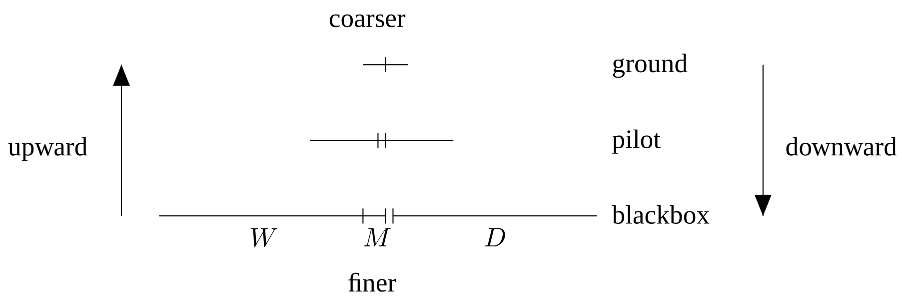

Figure 3.9: The air flight incident example.

Before turning to precisely define the granularity conversion, the assumptions underlying them must be clear. First of all, the considered language is qualitative and relational. Each layer represents a situation in the unaltered language of the relational algebra. This has the advantage of considering any description of a situation as being done under a particular granularity. Thus the layers are external to the language. The descriptions considered here are homogeneous (i.e., the language is the same for all the layers). The temporal structure is given by the algebra itself. The layers are organised as a partial order $\langle \mathcal{T}, \prec \rangle$ (sometimes it is known that a layer is coarser than another). In the example of Figure 3.9, it seems clear that $b \prec p \prec g$. It is not assumed that they are aligned or decomposed into homogeneous units, but the constraints below can enforce contiguity. The only operators considered here are the projection operators. The contextualisation operator is not explicit since (by opposition to logical systems) it cannot be composed with other operators. However, sometimes the notation ${}_g \rightarrow_{g'}$ is used, providing a kind of contextualisation (by specifying the concerned granularities). The displacement operator is useless since the relational language is not situated (or absolute, i.e., it does not evaluate the truth of a formula at a particular moment, but rather evaluates the truth of a temporal relationship between two entities).

3.5.2 Generic constraints on granularity change

Anyone can think about a particular set of projection operators by imagining the effects of coarseness. But here we provide a set of properties which should be satisfied by any system of granularity conversion operators. In fact, the set of properties is very small. Next section shows that they are sufficient for restricting the number of operators to only one (plus the expected operators corresponding to identity and conversion to everything).

Constraints below are given for unit relations (singletons of the set of relations). The operators on general relations are defined by:

$$\rightarrow R = \cup_{r \in R} \rightarrow r \tag{3.2}$$

Self-conservation

Self-conservation states that whatever be the conversion, a relationship must belong to its own conversion (this corresponds to the property named reflexivity when the conversion is a relation).

$$r \in\rightarrow r \qquad \text{(self-conservation)} \qquad (3.3)$$

It is quite a sensible and minimal property: the knowledge about the relationship can be less precise, but it must have a chance to be correct. Moreover, in a qualitative system, it is possible that nothing changes through granularity if the (quantitative) granularity step is small enough. Not requiring this property would disable the possibility that the same situation looks the same under different granularity. Self-conservation accounts for this.

Neighborhood compatibility

A property considered earlier is the *order preservation* property — stated in [Hobbs, 1985] as an equivalence: $\forall x, y, x < y \equiv (\rightarrow x) < (\rightarrow y)$. This property takes for granted the availability of an order relation (<) structuring the set of relationships. It states that

$$\text{if } x > y \text{ then } \neg(\rightarrow x <\rightarrow y) \qquad \text{(order preservation)}$$

However, order preservation has the shortcoming of requiring the order relation. Its algebraic generalization could be reciprocal avoidance:

$$\text{if } xry \text{ then } \neg(\rightarrow xr^{-1} \rightarrow y) \qquad \text{(reciprocal avoidance)}$$

Reciprocal avoidance is over-generalized and conflicts with self-conservation in case of auto-reciprocal relationships (i.e. such that $r = r^{-1}$). The neighborhood compatibility, while not expressed in [Euzenat, 1993], has been taken into account informally: it constrains the conversion of a relation to form a conceptual neighborhood (and hence the conversion of a conceptual neighborhood to form a conceptual neighborhood).

$$\forall r, \forall r', r'' \in\rightarrow r, \exists r_1, \dots r_n \in\rightarrow r :$$
$$r_1 = r', r_n = r'' \text{ and } \forall i \in [1, n-1] N(r_i, r_{i+1})$$
$$\text{(neighborhood compatibility)} \qquad (3.4)$$

This property has already been reported by Freksa [Freksa, 1992] who considers that a set of relationships must be a conceptual neighborhood in order to be seen as a coarse representation of the actual relationship. It is weaker than the two former proposals because it does not prevent the opposite to be part of the conversion. But in such a case, it constrains a path between the relation and its converse to be in the conversion too. Neighborhood compatibility seems to be the right property, partly because, instead of the former ones, it does not forbid a very coarse granularity under which any relationship is converted in the whole set of relations. It also seems natural because granularity can hardly be imagined as discontinuous (at least in continuous spaces).

Conversion-reciprocity distributivity

An obvious property for conversion is symmetry. It states that the conversion of the relation between a first object and a second one must be the reciprocal of the conversion of the relation between the second one and the first one. It is clear that the relationships between two temporal occurrences are symmetric and thus granularity conversion must respect this.

$$\rightarrow r^{-1} = (\rightarrow r)^{-1} \qquad \text{(distributivity of} \rightarrow \text{on } ^{-1}) \qquad (3.5)$$

Inverse compatibility

Inverse compatibility states that the conversion operators are consistent with each other, i.e., that if the relationship between two occurrences can be seen as another relationship under some granularity, then the inverse operation from the latter to the former can be achieved through the inverse operator. Stated otherwise, this property corresponds to symmetry when the operator is described as a relation.

$$r \in \bigcap_{r' \in \uparrow r} \downarrow r' \text{ and } r \in \bigcap_{r' \in \downarrow r} \uparrow r' \qquad \text{(inverse compatibility)} \qquad (3.6)$$

For instance, if someone in situation (p) of Figure 3.9 is able to imagine that, under a finer granularity (say situation b), there is some time between the misfiring period and the loss of altitude, then (s)he must be ready to accept that if (s)he were in situation (b), (s)he could imagine that there is no time between them under a coarser granularity (as in situation p).

Idempotency

A property which is usually considered first (especially in quantitative systems) is the full transitivity:

$$_g\rightarrow_{g'} \; _{g'}\rightarrow_{g''} r = {}_g\rightarrow_{g''} r \qquad \text{(transitivity)}$$

This property is too strong; it would for instance imply that:

$$_g\uparrow^{g'} \; ^{g'}\downarrow_g r = r$$

Of course, it cannot be achieved because this would mean that there is no loss of information through granularity conversion: this is obviously false. If it were true anyway, there would be no need for granularity operators: everything would be the same under any layer. On the other hand, other transitivity such as the oriented transitivity (previously known as cumulated transitivity) can be expected:

$$_g\uparrow^{g'}_{g'}\uparrow^{g''} r = {}_g\uparrow^{g''} r \text{ and } ^g\downarrow^{g'}_{g'}\downarrow_{g''} r = {}^g\downarrow_{g''} r \qquad \text{(oriented transitivity)}$$

However, in a purely qualitative calculus, the precise granularity (g) is not relevant and this property becomes a property of idempotency of operators:

$$\uparrow\uparrow r = \uparrow r \text{ and } \downarrow\downarrow r = \downarrow r \qquad \text{(idempotency)} \qquad (3.7)$$

At first sight, it could be clever to have non idempotent operators which are less and less precise with granularity conversion. However, if this applies very well to quantitative data, it does not apply for qualitative: the qualitative conversion applies equally for a large granularity conversion and for a small one which is ten times less. If, for instance, in a particular situation, a relationship between two entities is r, in a coarser representation it is r' and in an even coarser representation it is r'', then r'' must be a member of the upward conversion of r.

This is because r'' is indeed the result of a qualitative conversion from the first representation to the third. Thus, qualitatively, $\uparrow\uparrow=\uparrow$.

If there were no idempotency, converting a relationship directly would give a different result than when doing it through ten successive conversions.

Representation independence

Since the operation allowing one to go from a relational space to an interval relational space has been provided (by $\Leftarrow$ and $\Rightarrow$), the property constraining the conversion operators can also be given at that stage: representation independence states that the conversion must not be dependent upon the representation of the temporal entity (as an interval or as a set of bounding points). Again, this property must be required:

$$\rightarrow r = \Leftarrow\rightarrow\Rightarrow r \text{ and } \rightarrow r = \Rightarrow\rightarrow\Leftarrow r \qquad \text{(representation independence)} \qquad (3.8)$$

It can be though of as a distributivity:

$$\Rightarrow\rightarrow r = \rightarrow\Rightarrow r \text{ and } \Leftarrow\rightarrow r = \rightarrow\Leftarrow r$$

Note that, since $\Leftarrow$ requires that the relationship between bounding points allows the result to be an interval, there could be some restrictions on the results (however, these restrictions correspond exactly to the vanishing of an interval which is out of scope here).

The constraints (3.3, self-conservation) and (3.7, idempotence), together with the definition of the operators for full relations (3.2), characterise granularity operators as closure operators.

Nothing ensures that these constraints lead to a unique couple of operators for a given relational system.

Definition 3.5.1. *Given a relational system, a couple of operators up-down satisfying 3.3-3.7 is a coherent granularity conversion operator for that system.*

For any relation algebra there are two operators which always satisfy these requirements: the identity function (Id) which maps any relation into itself (or a singleton containing itself) and the non-informative function (Ni) which maps any relation into the base set of the algebra. It is noteworthy that these functions must then be their own inverse (i.e., they are candidates for both $\uparrow$ and $\downarrow$ at once). These solutions are not considered anymore below.

The framework provided so far concerns two operators related by the constraints, but there is no specificity of the upward or downward operator (this is why constraints are symmetric). By convention, if the system contains an equivalence relation (defined as e such that $e = e \circ e = e^{-1}$ [Hirsh, 1996]), the operators which maps this element to a strictly broader set is denoted as the downward operator. This meets the intuition because the coarser the view the more indistinguishable the entities (and they are then subject to the equivalence relation).

3.5.3 Results on point and interval algebras

From these constraints, it is possible to generate the possible operators for a particular relation algebra. This is first performed for the point algebra and the interval algebra in which it turns out that only one couple of non-trivial operators exists. Moreover, these operators satisfy the relationship between base and interval algebra.

Granularity for the point algebra

Proposition 3.5.1. *Table 3.1 defines the only possible non auto-inverse upward/downward operators for the point algebra.*

relation: r	$\uparrow r$	$\downarrow r$
$<$	$<=$	$<$
$=$	$=$	$<=>$
$>$	$>=$	$>$

Table 3.1: Upward and downward granularity conversions for the point algebra.

These operators fit intuition very well. For instance, if the example of Figure 3.9 is modeled through bounding points (x^- for the left endpoint and x^+ for the right endpoint) of intervals W^+, M^-, M^+ and D^-, it is represented in (b) by $W^+ = M^-$ (the engine stops working when it starts misfiring), $M^- < M^+$ (the beginning of the misfire is before its end), $M^+ < D^-$ (the end of the misfiring period is before the beginning of the loss of altitude) in (p) by $M^+ = D^-$ (the misfiring period ends when the loss of altitude begins) and in (g) by $M^- = M^+$ (the misfiring period does not exist anymore). This is possible by converting $M^+ < D^-$ into $M^+ = D^-$ ($= \in \uparrow <$) and $M^- = M^+$ into $M^- < M^+$ ($< \in \downarrow =$).

Granularity for the interval algebra

Since the temporal interval algebra is a plain interval algebra, the constraint 3.8 can be applied for deducing its granularity operators. This provides the only possible operators for the interval algebra. Table 3.2 shows the automatic translation from points to intervals:

r	$\uparrow r$				$\uparrow r$	$\downarrow r$				$\downarrow r$
b	$<=$	$<=$	$<=$	$<=$	bm	$<$	$<$	$<$	$<$	b
d	$>=$	$<=$	$>=$	$<=$	$dsfe$	$>$	$<$	$>$	$<$	d
o	$<=$	$<=$	$>=$	$<=$	$osmef^{-1}$	$<$	$<$	$>$	$<$	o
s	$=$	$<=$	$>=$	$=$	se	$<=>$	$<$	$>$	$<$	osd
f	$>=$	$<=$	$>=$	$=$	fe	$>$	$<$	$>$	$<=>$	$o^{-1}fd$
m	$<=$	$<=$	$=$	$<=$	m	$<$	$<$	$<=>$	$<$	bmo
e	$=$	$<=$	$>=$	$=$	e	$<=>$	$<$	$>$	$<=>$	$of^{-1}d^{-1}s$ $es^{-1}dfo^{-1}$

Table 3.2: Transformation of upward and downward operators between points into interval relation quadruples.

The conversion table for the interval algebra is given below. The corresponding operators enjoy the same properties as the operators for the point algebra.

Proposition 3.5.2. *The upward/downward operators for the interval algebra of Table 3.3 satisfy the properties 3.3 through 3.7.*

r	$\uparrow r$	$\downarrow r$	r^{-1}	$\uparrow r^{-1}$	$\downarrow r^{-1}$
b	bm	b	b^{-1}	$b^{-1}m^{-1}$	b^{-1}
d	$dfse$	d	d^{-1}	$d^{-1}s^{-1}f^{-1}e$	d^{-1}
o	$of^{-1}sme$	o	o^{-1}	$o^{-1}s^{-1}fem^{-1}$	o^{-1}
s	se	osd	s^{-1}	$s^{-1}e$	$d^{-1}s^{-1}o^{-1}$
f	fe	dfo^{-1}	f^{-1}	$f^{-1}e$	$d^{-1}f^{-1}o$
m	m	bmo	m^{-1}	m^{-1}	$o^{-1}m^{-1}b^{-1}$
e	e	$of^{-1}d^{-1}ses^{-1}dfo^{-1}$			

Table 3.3: Upward and downward granularity conversion for the interval algebra.

Proposition 3.5.3. *The upward/downward operators for the interval algebra of Table 3.3 are the only ones that satisfy the property 3.8 with regard to the operators for the point algebra of Table 3.1.*

If one wants to generate possible operators for the interval algebra, many of them can be found. But the constraint that this algebra must be the interval algebra (in the sense of [Hirsh, 1996]) of the point algebra restricts drastically the number of solutions.

The reader is invited to check on the example of Figure 3.9, that what has been said about point operators is still valid: the situation (b) is described by $W\{m\}M$ (the working period meets the misfiring one), $M\{b\}D$ (the misfiring period is anterior to the loss of altitude), in (p) by $M\{m\}D$ (the misfiring period meets the loss of altitude) and in (g) where the misfiring period does not appear anymore by $W\{m\}D$ (the working period meets the loss of altitude). This is compatible with the idea that, under a coarser granularity, b can become m ($m \in\uparrow b$) and that under a finer granularity m can become b ($b \in\downarrow m$).

The upward operator does not satisfy the condition 3.4 for B-neighborhood (in which objects are translated continuously [Freksa, 1992]) as it is violated by d, s, and f and C-neighborhood (in which the objects are continuously expanded or contracted by preserving their center of gravity [Freksa, 1992]) as it is violated by o, s, and f. This is because the corresponding neighborhoods are not based upon independent limit translations while this independence has been used for translating the results from the point algebra to the interval algebra.

It is noteworthy that the downward operator corresponds exactly to the closure of relationships that Ligozat [Ligozat, 1990] introduced in his own formalism. This seems natural since this closure, just like the conversion operators, provides all the adjacents relationships of a higher dimension.

3.5.4 General results of existence and composition

We provide here general results about the existence of granularity operators in algebra of binary relations. Then, the relationships between granularity conversion and composition, i.e., the impact of granularity changes on inference results, are considered.

Existence results for algebras of binary relations

The question of the general existence of granularity conversion operators corresponding to the above constraints can be raised. Concerning granularity conversion operators different from Id and Ni, two partial results have been established [Euzenat, 2001]. The first one shows that there are small algebras with no non-trivial operators:

Proposition 3.5.4. *The algebra based on two elements a and a^{-1} such that $N(a, a^{-1})$ has no granularity conversion operators other than identity and non-informative map.*

A more interesting result is that of the existence of operators for a large class of algebras. In the case of two auto-inverse operators (e.g., $=$ and $\neq$), there must exist conversion operators as shown by proposition 3.5.5. Proposition 3.5.5 exhibits a systematic way of generating operators from minimal requirements (but does not provide a way to generate all the operators). It only provides a sufficient, but not necessary, condition for having operators.

Proposition 3.5.5. *Given a relation algebra containing two relationships a and b such that $N(a, b)$ (it is assumed that neighborhood is converse independent, i.e., $N(a^{-1}, b^{-1})$), there exists a couple of upward/downward granularity operators defined by :*

if a and b are auto-inverse $\downarrow a = \{a, b\}$, $\uparrow b = \{a, b\}$, *the remainder being identity;*

if a only is auto-inverse $\downarrow a = \{a, b, b^{-1}\}$, $\uparrow b = \{a, b\}$, $\uparrow b^{-1} = \{a, b^{-1}\}$, *the remainder being identity;*

if a and b are not auto-inverse $\downarrow a = \{a, b\}$, $\uparrow b = \{a, b\}$, $\downarrow a^{-1} = \{a^{-1}, b^{-1}\}$, $\uparrow b^{-1} = \{a^{-1}, b^{-1}\}$, *the remainder being identity.*

There can be, in general, many possible operators for a given algebra. Proposition 3.5.5 shows that the five core properties of Section 3.5.2 are consistent. Another general question about them concerns their independence. It can be answered affirmatively:

Proposition 3.5.6. *The core properties of granularity operators are independent.*

This is proven by providing five systems satisfying all properties but one [Euzenat, 2001].

Granularity and composition

The composition of symbolic relationships is a favored inference means for symbolic representation systems. One of the properties which would be interesting to obtain is the independence of the results of the inferences from the granularity level (equation 3.9). The distributivity of $\rightarrow$ on $\circ$ denotes the independence of the inferences from the granularity under which they are performed.

$$\rightarrow (r \circ r') = (\rightarrow r) \circ (\rightarrow r') \qquad \text{(distributivity of } \rightarrow \text{ over } \circ) \qquad (3.9)$$

This property is only satisfied for upward conversion in the point algebra.

Proposition 3.5.7. *The upward operator for the point algebra satisfies property 3.9.*

It does not hold true for the interval algebra. Let three intervals x, y and z be such that xby and ydz. The application of composition of relations gives $x\{b\ o\ m\ d\ s\}z$ which, once upwardly converted, gives $x\{b\ m\ e\ d\ f\ s\ o\ f^{-1}\}z$. By opposition, if the conversion is first applied, it returns $x\{b\ m\}y$ and $y\{d\ f\ s\ e\}z$ which, once composed, yields $x\{b\ o\ m\ d\ s\}z$. The interpretation of this result is the following: by first converting, the information that there exists an interval y forbidding x to finish z is lost; however, if the relationships linking y to x and z are preserved, then the propagation will take them into account and recover the lost precision: $\{b\,m\,e\,d\,f\ s\ o^{\ -1}\}\circ\{b\,o\,m\,d\,s\} = \{b\,o\,m\,d\,s\}$. In any case, this cannot be enforced since, if the length of y is so small that the conversion makes it vanish, the correct information at that granularity is the one provided by applying first the composition: x can meet the end of z under such a granularity. However, if equation 3.9 cannot be achieved for upward conversion in the interval algebra, upward conversion is super-distributive over composition.

Proposition 3.5.8. *The upward operator for the interval algebra satisfies the following property:*

$$(\uparrow r)\circ(\uparrow r') \subseteq \uparrow (r\circ r') \qquad \text{(super-distributivity of } \uparrow \text{ over } \circ)$$

A similar phenomenon appears with the downward conversion operators (it appears both for points and intervals). Let x, y and z be three points such that $x > y$ and $y = z$. On the one hand, the composition of relations gives $x > z$, which is converted to $x > z$ under the finer granularity. On the other hand, the conversion gives $x > y$ and $y{<}{=}{>}z$ because, under a more precise granularity, y could be close but not really equal to z. The composition then provides no more information about the relationship between x and z ($x{<}{=}{>}z$). This is the reverse situation as before: it takes into account the fact that the non-distinguishability of two points cannot be ensured under a finer grain. Of course, if everything is converted first, then the result is as precise as possible: downward conversion is sub-distributive over composition.

Proposition 3.5.9. *The downward operators for the interval and point algebras satisfy the following property:*

$$\downarrow (r\circ r') \subseteq (\downarrow r)\circ(\downarrow r') \qquad \text{(sub-distributivity of } \downarrow \text{ over } \circ)$$

These two latter properties can be useful for propagating constraints in order to get out of them the maximum of information quickly. For instance, in the case of upward conversion, if no interval vanishes, every relationship must be first converted and then composed.

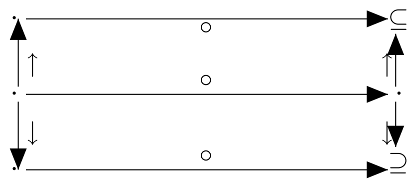

Figure 3.10: A diagrammatic summary of Propositions 3.5.9 and 3.5.8.

These properties have been discovered independently in the qualitative case [Euzenat, 1993] and in the set-theoretic granularity area through an approximation algorithm for quantitative constraints [Bettini *et al.*, 1996].

3.5.5 Granularity through discrete approximation

The algebra of relations can be directly given or derived as an interval algebra. It can also be provided by axiomatizing properties of objects or generated from properties of artefacts. Bittner [Bittner, 2002] has taken such an approach for generating sets of relations depending on the join of related objects. He has adapted a framework for qualitatively approximating spatial position to temporal representation. This framework can be used in turn for finding approximate relations between temporal entities which can be seen as relations under a coarser granularity.

Qualitative temporal relations

This work is based on a new analysis of the generation of relations between two spatial areas. These relations are characterized through the ”intersection” (or meet) between the two regions. More precisely, the relation is characterized by the triple:

$$\langle x \wedge y \not\approx \perp, x \wedge y \approx x, x \wedge y \approx y \rangle$$

The items in these triples characterize the non emptiness of $x \wedge y$ (1st item) and its relation to x and y (2nd and 3rd items). So the values of this triple are relations (this approach is inspired from [Egenhofer and Franzosa, 1991]). These values are taken out of a set of possible relations Ω. This generates several different sets of relations depending on the kind of relations used:

- boundary insensitive relations (RCC5);
- one-dimensional boundary insensitive relations between intervals (RCC_1^9);
- one-dimensional boundary insensitive relations between non convex regions (RCC_1^9);
- boundary sensitive relations (RCC8);
- one-dimensional boundary sensitive relations (RCC_1^{15}).

Some of these representations are obviously refinement of others. In that sense, we obtain a granular representation of a temporal situation by using more or less precise qualitative relationships. This can also be obtained by using other kinds of temporal representations (RCC8 is less precise than Allen’s algebra of relations).

As an example, RCC_1^9 considers regions x and y corresponding to intervals on the real line. The set Ω is made of FLO, FLI, T, FRI, FRO. FLO indicates that no argument is included in the other (O) and there is some part of the first argument left (L) of the second one, FLI indicates that the second argument is included in the first one and there is some part of the first argument left (L) of the second one, T corresponds to the equality of the intersection with the interval, and FRI and FRO are the same for the right hand bound. This provides the relations of Table 3.4.

$x \wedge y \not\sim \bot$	$x \wedge y \sim x$	$x \wedge y \sim y$	Allen
FLO	FLO	FLO	b m
FRO	FRO	FRO	b^{-1} m^{-1}
T	FLO	FLO	o
T	FRO	FRO	o^{-1}
T	T	FLI	d s
T	T	FRI	d f
T	FLI	T	d^{-1} f^{-1}
T	FRI	T	d^{-1} s^{-1}
T	T	T	e

Table 3.4: The relations of RCC_1^9.

The relations in these sets are not always jointly exhaustive and pairwise disjoint. For instance, RCC_1^9 is exhaustive but not pairwise disjoint, simply because d and d^{-1} appear in two lines of the table.

Qualitative temporal locations

The framework as it is developed in [Bittner and Steel, 1998] considers a space, here a temporal domain, as a set of places T_0. Any spatial or temporal occurrence will be a subset of T_0. So, with regard to what has been considered in Section 3.3, the underlying space is aligned and structured.

An approximation is based on the partition of T_0 into a set of cells K (i.e., $\forall k, k' \in K, k \subseteq T_0, k \cap k' = \emptyset$ and $\cup_{k \in K} k = T_0$). The localization of any temporal occurrence is then approximated by providing its relation to each cell. The location of $x \subseteq T_0$ is a function $\rho_x : K \rightarrow \Omega'$ from the set of cells to a set of relations Ω' (which may but have not to correspond to Ω or a RCC_q^p defined above). The resulting approximation is thus dependent on the partition K and the set of relations Ω'.

From this, we can state that two occurrences x and y are indistinguishable under granularity $\langle K, \Omega' \rangle$ if and only if $\rho_x = \rho_y$. This formulation is typical from the set-theoretic approach to temporal granularity used in a strictly qualitative domain.

We can also define the interpretation of an area of the set of cells ($X : K \rightarrow \Omega$) as the set of places it approximates:

$$[X] = \{x \subseteq T_0 | \rho_x = X\}$$

Relations between approximations and granularity

It is clear that the approximation of a region x can be considered as its representation $\uparrow x$ under the granularity $\langle K, \Omega' \rangle$ (i.e., ρ_x). In the same vein, the interpretation of approximation $[X]$ corresponds to the conversion of this region to the finer granularity $\downarrow X$. In that respect we are faced with two discrete and aligned granularities.

The following question can be raised: given a relation $r \in RCC_q^p$ between x and y, the approximations $\uparrow x$ and $\uparrow y$, and $\uparrow r$ holding between $\uparrow x$ and $\uparrow y$, what can be said of the relationship between r and $\uparrow r$? The approximate relation $\uparrow r$ holding between X and Y is characterized as $SEM(X,Y)$ and defined as:

$$SEM(X,Y) = \{r \in RCC_q^p | x \in [X], y \in [Y], xRy, \text{ and } r \in R\}$$

The author goes on to define a syntactic operator ($SYN(X,Y)$) for determining the relationships between approximate regions. This operator must be as close as possible to $SEM(X,Y)$. It is defined by replacing in the equations defining the relations of the considered set, the region variables (x and y) by approximation variables (X and Y) and the meet operation by upper or lower bounds for the meet operation. This provides a pair of values for the relations between X and Y depending on whether they have been computed with the upper and lower meet.

It is now possible to obtain the relations between granular representations of the entities by considering that $x \uparrow r\ y$ can be obtained in the usual way (but for obtaining $\uparrow r$ we need to consider all the possible granularities, i.e., all the possible K and all the possible Ω'). $X \downarrow r\ Y$ is what should be obtained by $SEM(X,Y)$ and approximated by $SYN(X,Y)$.

Hence, a full parallel can be made between the above-described work on qualitative granularity and this work on discrete approximation in general. Unfortunately, the systems developed in [Bittner, 2002] do not include Allen's algebra. The satisfaction of the axioms by this scheme has not been formally established. However, one can say that self-conservation and idempotence are satisfied. Neighborhood compatibility depends on a neighborhood structure, but $SYN(X,Y)$ is very often an interval in the graph of relations (which is not very far from a neighborhood structure). It could also be interesting to show that when RCC_1^{15} relations correspond to Allen's ones, the granularity operators correspond.

In summary, this approximation framework has the merit of providing an approximated representation of temporal places interpreted on the real line. The approximation operation itself relies on aligned granularities. This approach is entirely qualitative in its definition but can account for orientation and boundaries.

3.6 Applications of time granularity

Time granularity come into play in many classes of applications with different constraints. Thus, the contributions presented below not only offer an application perspective, but generally provide their own granular formalism. The fact that there are no applications to multi-agent communication means that the agents currently developed communicate with agents of the same kind. With the development of communicating programs, it will become necessary to consider the compatibility of two differently grained descriptions of what they perceive.

3.6.1 Natural language processing, planning, and reasoning

The very idea of granularity in artificial intelligence comes from the field of natural language understanding [Hobbs, 1985]. In [Gayral, 1992] Gayral and Grandemange take into account the same temporal unit under a durative or instantaneous aspect. Their work is motivated by problems in text understanding. A mechanism of upward/downward conversion is introduced and modeled in a logical framework. It only manages symbolic constraints and it converts the entities instead of their relationships. The representation they propose is based on a notion of composition and it allows the recursive decomposition of beginning and ending bounds of intervals into new intervals. The level of granularity is determined during text understanding by the election of a distinguished individual (which could be compared with a focus of attention) among the set of entities and the aspect (durative vs. instantaneous) of that individual. Unlike most of the previously-described approaches, where granularity is considered orthogonal to a knowledge base, in Gayral and Grandemange's work the current granularity is given relatively to the aspect of a particular event. A link between the two notions can be established by means of the decomposition relation between entities (or history [Euzenat, 1993]). Time granularity in natural language processing and its relation with the durative/instantaneous aspects have been also studied by other authors. As an example, Becher et al. model granularity by means of time units and two basic relations over them: precedence and containment (alike the set-theoretic approach, Section 3.3) [Becher *et al.*, 1998]. From a model of time units consisting of a finite sequence of rational numbers, the authors build an algebra of relations between these units, obtaining an algebraic account of granularity.

In [Badaloni and Berati, 1994], Badaloni and Berati use different time scales in an attempt to reduce the complexity of planning problems. The system is purely quantitative and it relies on the work presented in Section 3.3. The NatureTime [Mota *et al.*, 1997] system is used for integrating several ecological models in which the objects are modeled under different time scales. The model is quantitative and it explicitly defines (in Prolog) the conversions from a layer to another. This is basically used during unification when the system unifies the temporal extensions of the atoms. Combi et al. [Combi *et al.*, 1995] applied their multi-granular temporal database to clinical medicine. The system is used for the follow-up of therapies in which data originate from various physicians and the patient itself. It allows one to answer (with possibility of undefined answers) to various questions about the history of the patient. In this system (like in many other) granularity usually means "converting units with alignment problems".

3.6.2 Program specification and verification

In [Ciapessoni *et al.*, 1993], Ciapessoni et al. apply the logics of time granularity to the specification and verification of real-time systems. The addition of time granularity makes it possible to associate coarse granularities with high-level modules and fine granularities with the lower level modules that compose them. In [Fiadeiro and Maibaum, 1994], Fiadeiro and Maibaum achieve the same practical goal by considering a system in which granularity is defined a posteriori (it corresponds to the granularity of actions performed by modules, while in the work by Ciapessoni et al. the granularity framework is based on a metric time) and the refinement (granularity change) takes place between classical logic theories instead of inside a specialized logical framework (as in Section 3.4.1). It is worth pointing out that both contributions deal with refinement, in a quite different way, but they do not take into account upward granularity change. Finally, in [Broy, 1997], Broy introduces the notion of temporal refinement into the description of software components in such a way that the behavior of these components is temporally described under a hierarchy of temporal models.

3.6.3 Temporal Databases

Time granularity is a long-standing issue in the area of temporal databases (see Chapter 14). As an evidence of the relevance of the notion of time granularity, the database community has released a "glossary of time granularity concepts" [Bettini *et al.*, 1998a]. As we already pointed out, the set-theoretic formalization of granularity (see Section 3.3) has been settled in the database context. Moreover, besides theoretical advances, the database community contributed some meaningful applications of time granularity. As an example, in [Bettini *et al.*, 1998b] Bettini et al. design an architecture for dealing with granularity in federated databases involving various granularities. This work takes advantage of extra information about the database design assumptions in order to characterize the required transformations. The resulting framework is certainly less general than the set-theoretic formalization of time granularity reported in Section 3.3, but it brings granularity to concrete databases applications. Time granularity has also been applied to data mining procedures, namely, to procedures that look for repeating collection of events in federated databases [Bettini *et al.*, 1998d] by solving simple temporal reasoning problems involving time granularities (see Section 3.3). An up-to-date account of the system is given in [Bettini *et al.*, 2003].

3.6.4 Granularity in space

(Spatial) granularity plays a major role in geographic information systems. In particular, the granularity for the Region Connection Calculus [Randell *et al.*, 1992; Egenhofer and Franzosa, 1991] has been presented in that context [Euzenat, 1995b]. Moreover, the problem of generalization is heavily related to granularity [Muller *et al.*, 1995]. Generalization consists in converting a terrain representation into a coarser map. This is the work of cartographers, but due to the development of computer representation of the geographic information, the problem is now tackled in a more formal, and automated, way.

In [Topaloglou, 1996], Topaloglou et al. have designed a spatial data model based on points and rectangles. It supports aligned granularities and it is based on numeric constraints. The treatment of granularity consists in tolerant predicates for comparing objects of different granularities which allow two objects to be considered as equals if they only deviate from the granularity ratio.

In [Puppo and Dettori, 1995; Dettori and Puppo, 1996], Puppo and Dettori outline a general approach to the problem of spatial granularity. They represent space as a cell complex (a set of elements with a relation of containment and the notion of dimension as a map to integers) and generalization as a surjective mapping from one complex cell into another. One can consider the elements as simplexes (points of dimension 1, segments of dimension 2 bounded by two points, and triangles of dimension 3 bounded by three segments). This notion of generalization takes into account the possible actions on an object: preservation, if it persists with the same dimension under the coarser granularity, reduction, if it persists at a lower dimension, and immersion, if it disappears (it is then considered as immersed in another object). The impact of these actions on the connected objects is also taken into account through a set of constraints, exactly like it has been done in Section 3.5.2. This should be totally compatible with the two presentations of granularity given here. Other transformations, such as exaggeration (when a road appears larger than it is under the map scale) and displacement, have been taken into account in combination with generalization, but they do not fit well in the granularity framework given in Section 3.2. Last, it must be noted that these definitions are only algebraic and that no analytical definitions of the transformations have been given.

Other authors have investigated multi-scale spatial databases, where a simplified version of the alignment problem occurs [Rigaux and Scholl, 1995]. It basically consists in the requirement that each partition of the space is a sub-partition of those it is compared with (a sort of spatial alignment).

Finally, some implementations of multi-resolution spatial databases have been developed with encouraging results [Devogele *et al.*, 1996]. As a matter of fact, the addressed problem is simpler than that of generalization, since it consists in matching the elements of two representations of the same space under different resolutions. While generalization requires the application of a (very complex) granularity change operator, this problem only requires to look for compatibility of representations. Tools from databases and generalization can be used here.

3.7 Related work

We would like to briefly summarize the links to time granularity coming from a variety of research fields and to provide some additional pointers to less-directly related contributions which have not been fully considered here due to the lack of space. Relationships with research in databases have been discussed in Sections 3.3 and 3.6.3. Granularity as a phenomenon that affects space has been considered in Section 3.6.4. The integration of a notion of granularity into logic programming is dealt with in [Mota *et al.*, 1997; Liu and Orgun, 1997] (see Section 3.6.1 and see also Chapter 13). Work in qualitative reasoning can also be considered as relevant to granularity [Kuipers, 1994] (see Chapter 20).

The relationships between (time) granularity and formal tools for abstraction have been explored in various papers. As an example, Giunchiglia et al. propose a framework for abstraction which applies to a structure $\langle L, A, R\rangle$, where L is a language, A is a set of axioms, and R is a set of inference rules [Giunchiglia *et al.*, 1997]. They restrict abstraction to A, because the granularity transformations are constrained to remain within the same language and the same rules apply to any abstraction. One distinctive feature of this work is that it is oriented towards an active abstraction (change of granularity) in order to increase the performance of a system. As a matter of fact, using a coarse representation reduces the problem size by getting rid of details. The approaches to time granularity we presented in this chapter are more oriented towards accounting for the observed effects of granularity changes instead of creating granularity change operators which preserve certain properties.

Concluding remarks

We would like to conclude this chapter by underlining the relevance and complexity of the notion of time granularity. On the one hand, when some situations can be seen from different viewpoints (of designers, observers, or agents), it is natural to express them under different granularities. On the other hand, problems immediately arise from using multiple granularity, because it is difficult to assign a proper (or, at least, a consistent) meaning to these granular representations.

As it can be seen from above, a lot of work has already been devoted to granularity. This research work has been developed in various domains (e.g., artificial intelligence, databases, and formal specification) with various tools (e.g., temporal logic, set theory, and algebra of relations). It must be clear that the different approaches share many concepts and results, but they have usually considered different restrictions. The formal models have provided constraints on the interpretations of the temporal statements under a particular granularity, but they did not provide an univocal way to interpret them in a specific application context.

On the theoretical side, further work is required to formally compare and/or integrate the various proposals. On the application side, if the need for granularity handling is acknowledged, it is not very developed in the solutions. There are reasons to think that this will change in the near future, drained by applications such as federated databases and agent systems, providing new problems to theoretical research.

Bibliography

[Åqvist, 1979] L. Åqvist. A Conjectured Axiomatization of Two-Dimensional Reichenbachian Tense Logic. *J. Philosophical Logic*, 8:1–45, 1979.

[Abadi and Manna, 1985] M. Abadi and Z. Manna. Nonclausal Temporal Deduction. *Lecture Notes in Computer Science*, 193:1–15, 1985.

[Abadi and Manna, 1989] M. Abadi and Z. Manna. Temporal Logic Programming. *Journal of Symbolic Computation*, 8(3), 1989.

[Abadi and Manna, 1990] M. Abadi and Z. Manna. Nonclausal Deduction in First-Order Temporal Logic. *Journal of the ACM*, 37(2):279–317, 1990.

[Abadi, 1987] M. Abadi. *Temporal-Logic Theorem Proving*. PhD thesis, Department of Computing, Stanford University, 1987. STAN-CS-87-1151.

[Abiteboul and Grumbach, 1988] S. Abiteboul and S. Grumbach. A Logic-Based Language for Complex Objects. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*, 1988.

[Abiteboul *et al.*, 1991] S. Abiteboul, P. Kanellakis, and G. Grahne. On the Representation and Querying of Sets of Possible Worlds. *Theoretical Computer Science*, 78(1):159–187, 1991.

[Abiteboul *et al.*, 1995] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

[Abiteboul *et al.*, 1996] S. Abiteboul, L. Herr, and J. Van den Bussche. Temporal Versus First-Order Logic to Query Temporal Databases. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 49–57, 1996.

[Abiteboul *et al.*, 1999] S. Abiteboul, L. Herr, and J. Van den Bussche. Temporal Connectives Versus Explicit Timestamps to Query Temporal Databases. *Journal of Computer and System Sciences*, 58(1):54–68, 1999.

[Ahn, 1986] I. Ahn. Towards an Implementation of Database Management Systems with Temporal Support. In *Proceedings of the 2nd International Conference on Data Engineering*, pages 374–381, 1986.

[Alferes and Pereira, 1996] J. Alferes and L. Pereira. *Reasoning With Logic Programming*. Springer Verlag, 1996.

[Aliferis and Cooper, 1996] C. Aliferis and G. Cooper. A Structurally and Temporally Extended Bayesian Belief Network Model: Definitions, Properties, and Modeling Techniques. In *Proceedings of International Conference on Uncertainty in AI (UAI)*, pages 28–38, 1996.

[Allen and Ferguson, 1994] J.F. Allen and G. Ferguson. Actions and events in interval temporal logic. *Journal of Logic and Computation*, 4(5):531–579, October 1994.

[Allen and Hayes, 1985] J. Allen and P. Hayes. A Common-Sense Theory of Time. In *Proceedings of the Nineth International Joint Conference on Artificial Intelligence (IJCAI-85)*, pages 528–531, Los Angeles CA, USA, 1985. Morgan Kaufmann.

[Allen and Hayes, 1989] J. Allen and P. Hayes. Moments and Points in an Interval-Based Temporal Logic. *Computational Intelligence*, 5(4):225–238, November 1989.

[Allen, 1983] J. F. Allen. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[Allen, 1984] James Allen. Towards a General Theory of Action and Time. *Artificial Intelligence*, 23:123–154, 1984.

[Allen, 1991a] James F. Allen. Temporal reasoning and planning. In J. F. Allen, H. Kautz, R. N. Pelavin, and J. Tenenberg, editors, *Reasoning about Plans*, chapter 1, pages 1–67. Morgan Kaufmann, 1991.

[Allen, 1991b] J.F. Allen. Time and Time Again: The Many Ways to Represent Time. *International Journal of Intelligent Systems*, 6(4):341–356, July 1991.

[Alur and Henzinger, 1991] R. Alur and T.A. Henzinger. Logics and Models of Real-Time: A Survey. In *Real Time: Theory in Practice*, volume 600 of *Lecture Notes in Computer Science*, pages 74–106. Springer-Verlag, 1991.

[Alur and Henzinger, 1993] R. Alur and T. Henzinger. Real-Time Logics: Complexity and Expressiveness. *Information and Computation*, 104:35–77, 1993.

[Alur *et al.*, 1993] R. Alur, C. Courcoubetis, T. Henzinger, and P. Ho. Hybrid Automata: an Algorithmic Approach to the Specification and Analysis of Hybrid Systems. In *Proceedings of International Workshop on Theory of Hybrid Systems*, pages 209–229, 1993.

[Apt and Pellegrini, 1994] K. Apt and A. Pellegrini. On the Occur-Check Free Logic Programs. *ACM Transactions on Programming Languages and Systems*, 16(3):687–726, 1994.

[Apt *et al.*, 1988] K.R. Apt, H.A. Blair, and A. Walker. Towards a Theory of Declarative Knowledge. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 89–148. Morgan Kaufmann, 1988.

[Apt, 1990] K.R. Apt. Logic Programming. In Jan van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 10, pages 493–574. Elsevier/MIT Press, 1990.

[Arasu *et al.*, 2002] A. Arasu, B. Babcock, S. Babu, J. McAlister, and J. Widom. Characterizing Memory Requirements for Queries over Continuous Data Streams. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 221–232, 2002.

[Artale and Franconi, 1994] A. Artale and E. Franconi. A Computational Account for a Description Logic of Time and Action. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourth International Conference (KR'94)*, pages 3–14, San Francisco, CA, 1994. Morgan Kaufmann.

[Artale and Franconi, 1998] A. Artale and E. Franconi. A Temporal Description Logic for Reasoning about Actions and Plans. *Journal of Artificial Intelligence Research*, 9:463–506, 1998.

[Artale and Franconi, 1999] A. Artale and E. Franconi. Temporal Entity-Relationship Modeling with Description Logics. In *Proceedings of the International Conference on Conceptual Modeling (ER'99)*. Springer-Verlag, November 1999.

[Artale and Franconi, 2000] A. Artale and E. Franconi. Temporal Description Logics for Conceptual Modelling, July 2000. Technical report, Department of Computer Science, University of Manchester, UK.

[Artale and Franconi, 2001] A. Artale and E. Franconi. A Survey of Temporal Extensions of Description Logics. *Annals of Mathematics and Artificial Intelligence*, 30(1-4), 2001.

[Artale and Lutz, 1999] A. Artale and C. Lutz. A Correspondence between Temporal Description Logics. In *Proceedings of the 1999 Description Logic Workshop (DL'99)*, pages 145–149, 1999.

[Aylett *et al.*, 1998] R. Aylett, J. Soutter, G. Petley, and P. Chung. AI planning in a chemical plant domain. In *Proceedings of European Conference on Artificial Intelligence (ECAI)*, pages 622–626, 1998.

[Baader and Hanschke, 1991] F. Baader and P. Hanschke. A Scheme for Integrating Concrete Domains into Concept Languages. In *Proceedigns of Twelfth international Conference on Artificial Intelligence (IJCAI)*, pages 446–451, Sidney, Australia, 1991.

[Baader and Hanschke, 1992] F. Baader and P. Hanschke. Extensions of Concept Languages for a Mechanical Engineering Application. In *Proceedings of the 16th German AI-Conference (GWAI-92)*, volume 671 of *Lecture Notes in Computer Science*, pages 132–143. Springer-Verlag, 1992.

[Baader and Ohlbach, 1995] F. Baader and H-J. Ohlbach. A Multi-Dimensional Terminological Knowledge Representation Language. *Journal of Applied Non-Classical Logics*, 5:153–19, 1995.

[Babcock *et al.*, 2002] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and Issues in Data Stream Systems. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 1–16, 2002.

[Bacchus and Ady, 2001] F. Bacchus and M. Ady. Planning with Resources and Concurrency: A Forward Chaining Approach. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 417–424, 2001.

[Bacchus and Kabanza, 1996] F. Bacchus and F. Kabanza. Planning for Temporally Extended Goals. In *Proceedings of AAAI'96*, pages 1215–1222. AAAI press, 1996.

[Bacchus and Kabanza, 1998] F. Bacchus and F. Kabanza. Planning for Temporally Extended Goals. *Annals of Mathematics and Artificial Intelligence*, 22:5–27, 1998.

[Bacchus and Kabanza, 2000] F. Bacchus and F. Kabanza. Using Temporal Logic to Express Search Control Knowledge for Planning. *Artificial Intelligence*, 116(1-2):123–191, 2000.

[Bacchus *et al.*, 1991] F. Bacchus, J. Tenenberg, and J. Koomen. A Non-Reified Temporal Logic. *Artificial Intelligence*, 52(1):87–108, 1991.

[Bacchus, 2001] F. Bacchus. The AIPS'00 Planning Competition. *AI Magazine*, 22(3):47–56, 2001.

[Bach, 1986] E. Bach. The Algebra of Events. *Linguistics and Philosophy*, 9:5–16, 1986.

[Badaloni and Berati, 1994] S. Badaloni and M. Berati. Dealing with Time Granularity in a Temporal Planning System. *Lecture Notes in Computer Science*, 827:101–116, 1994.

[Baioletti *et al.*, 2000] M. Baioletti, S. Marcugini, and A. Milani. DPPlan: An Algorithm for Fast Solutions Extraction from a Planning Graph. In *Proceedings of AIPS*, pages 13–21, 2000.

[Baker, 1989] A.B. Baker. A simple solution to the Yale Shooting Problem. In R.J. Brachman, H.J. Levesque, and R. Reiter, editors, *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning (KR-89)*, pages 11–20. Morgan Kaufmann, 1989.

[Balbiani *et al.*, 1998] P. Balbiani, J.-F. Condotta, and L.F. del Cerro. Bidimensional Temporal Relations. In *Proceedings of International Conference on Knowledge Representation and Reasoning (KR)*, June 1998.

[Balbiani *et al.*, 2003] P. Balbiani, J.F. Condotta, and G. Ligozat. On the Consistency Problem for the indu Calculus. In *Proceedings of the 10th International Symposium on Temporal Representation and Reasoning and Fourth International Conference on Temporal Logic (TIME-ICTL'03)*. IEEE Computer Society, 2003.

[Balduccini *et al.*, 2000] M. Balduccini, M. Gelfond, and M. Nogueira. A-Prolog as a Tool for Declarative Programming. In *Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, 2000.

[Banieqbal and Barringer, 1986] B. Banieqbal and H. Barringer. A Study of an Extended Temporal Logic and a Temporal Fixed Point Calculus. Technical Report UMCS-86-10-2, University of Manchester, Manchester, October 1986.

[Banieqbal and Barringer, 1987] B. Banieqbal and H. Barringer. Temporal logic with fixed points. In *Temporal Logic in Specification*, page 62. LNCS 398 Springer-Verlag, April 1987.

[Barahona, 1994] P. Barahona. A Causal and Temporal Reasoning Model and its use in Drug Therapy Applications. *Artificial Intelligence in Medicine*, 6:1–27, 1994.

[Baral and Gelfond, 1994] C. Baral and M. Gelfond. Logic programming and knowledge representation. *Journal of Logic Programming*, 19,20:73–148, 1994.

[Baral and Gelfond, 1997] C. Baral and M. Gelfond. Reasoning about Effects of Concurrent Actions. *Journal of Logic Programming*, 31(1-3):85–117, May 1997.

[Baral and Gelfond, 2000] C. Baral and M. Gelfond. Reasoning Agents in Dynamic Domains. In J. Minker, editor, *Logic Based Artificial Intelligence*. Kluwer, 2000.

[Baral *et al.*, 1997] C. Baral, M. Gelfond, and A. Provetti. Representing Actions: Laws, Observations and Hypothesis. *Journal of Logic Programming*, 31(1-3):201–243, May 1997.

[Baral, 1995] C. Baral. Reasoning about Actions : Non-deterministic effects, Constraints and Qualification. In *Proceedings of International Joint Conference on Artificial Intelligenc (IJCAI)*, pages 2017–2023, 1995.

[Baral, 1997] C. Baral. Embedding Revision Programs in Logic Programming Situation Calculus. *Journal of Logic Programming*, 30(1):83–97, Jan 1997.

[Baral, August 1994] C. Baral. Rule Based Updates on Simple Knowledge Bases. In *Proceedings of AAAI'94*, pages 136–141, August 1994.

[Barber, 1993] F. A. Barber. A Metric Time-Point and Duration-Based Temporal Model. *SIGART Bulletin*, 4(3):30–49, 1993.

[Barendregt, 1984] H. P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, 1984.

[Barnes and Barnett, 1995] M. Barnes and G. Barnett. An Architecture for a Distributed Guideline Server. In R. M. Gardner, editor, *Proceedings of the Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pages 233–237, New Orleans, USA, 1995. Hanley & Belfus.

[Barringer and Kuiper, 1984] H. Barringer and R. Kuiper. Hierarchical Development of Concurrent Systems in a Temporal Logic Framework. In S.D. Brookes, A.W. Roscoe, and G. Winskel, editors, *Proceedings of the NSF/SERC Seminar on Concurrency*, volume 197 of *Lecture Notes in Computer Science*, pages 35–61. Springer-Verlag, Heidleberg, 1984.

[Barringer *et al.*, 1984] H. Barringer, R. Kuiper, and A. Pnueli. Now You May Compose Temporal Logic Specifications. In *Proceedings of the 16th Symposium on Theory of Computing (STOC)*, pages 51–63. ACM, April 1984.

[Barringer *et al.*, 1986] H. Barringer, R. Kuiper, and A. Pnueli. A really abstract concurrent model and its temporal logic. In *Proceedings of the Thirteenth ACM Symposium on the Principles of Programming Languages*, pages 173–183, St. Petersberg Beach, Florida, January 1986. ACM Press.

[Barringer *et al.*, 1995] H. Barringer, M. Fisher, D. Gabbay, G. Gough, and R. Owens. METATEM: An Introduction. *Formal Aspects of Computing*, 7(5):533–549, 1995.

[Barringer *et al.*, 1996] H. Barringer, M. Fisher, D. Gabbay, R. Owens, and M. Reynolds, editors. *The Imperative Future: Principles of Executable Temporal Logics*. Research Studies Press, Chichester, United Kingdom, 1996.

[Baudinet *et al.*, 1993] M. Baudinet, J. Chomicki, and P. Wolper. Temporal Deductive Databases. In Tansel et al. [1993], pages 294–320.

[Baudinet *et al.*, 1999] M. Baudinet, J. Chomicki, and P. Wolper. Constraint-Generating Dependencies. *Journal of Computer and System Sciences*, 59(1):94–115, 1999.

[Baudinet, 1992] M. Baudinet. A Simple Proof of the Completeness of Temporal Logic Programming. In L. Fariñas del Cerro and M. Penttonen, editors, *Intensional Logics for Programming*. Oxford University Press, 1992.

[Baudinet, 1995] M. Baudinet. On the Expressiveness of Temporal Logic Programming. *Information and Computation*, 117(2):157–180, 1995.

[Becher *et al.*, 1998] G. Becher, F. Clérin-Debart, and P. Enjalbert. A Model for Time Granularity in Natural Language. In *Proceedings of International Workshop on Temporal Representation and Reasoning (TIME)*, Los Alamitos (CA US), 1998. IEEE computer society press.

[Bell and Tate, 1985] C.E. Bell and T. Tate. Use and Justification of Algorithms for Managing Temporal Knowledge in o-plan. Technical Report 531, AIAI, Edinburgh, U.K., 1985.

[Bellini *et al.*, 2000] P. Bellini, R. Mattolini, and P. Nesi. Temporal Logics for Real-Time System Specification. *ACM Computing Surveys*, 32(1):12–42, March 2000.

[Bench-Capon *et al.*, 1988] T. Bench-Capon, G. Robinson, T. Routen, and M. Sergot. Logic Programming for Large Scale Applications in Law: A Formalization of Supplementary Benefit Legislation. In Hayes, Michie, and Richards, editors, *Machine Intelligence*, pages 209–260. Oxford Univ. Press, 1988.

[Benerecetti *et al.*, 1998] M. Benerecetti, F. Giunchiglia, and L. Serafini. Model Checking Multiagent Systems. *Journal of Logic and Computation*, 8(3):401–423, 1998.

[Bennett *et al.*, 2002a] B. Bennett, A. Cohn, F. Wolter, and M. Zakharyaschev. Multi-dimensional modal logic as a framework for spatio-temporal reasoning. *Applied Intelligence*, 2002.

[Bennett *et al.*, 2002b] B. Bennett, C. Dixon, M. Fisher, E. Franconi, I. Horrocks, and M. de Rijke. Combinations of Modal Logics. *AI Review*, 17(1):1–20, 2002.

[Bennett, 1988] J. Bennett. *Events and Their Names*. Clarendon Press, Oxford, 1988.

[Benthem, 1984] J.F.A.K. van Benthem. Correspondence Theory. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic - volume II*. Reidel, Dordrecht, 1984.

[Benthem, 1988a] J.F.A.K. van Benthem. *A Manual of Intensional Logic - Second Edition, Revised and Expanded*, volume 1 of *CSLI Lecture Notes*. Center for the Study of Language and Information, Stanford University, California, 1988.

[Benthem, 1988b] J.F.A.K. van Benthem. Time, Logic and Computation. In J. W. de Bakker, W.-P. de Roever, and G. Rozenberg, editors, *Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency - LNCS Vol. XXX*, pages 1–49. Springer-Verlag, Heidelberg, June 1988.

[Benthem, 1995] J.F.A.K. van Benthem. Temporal logic. In D. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming, Volume 4: Epistemic and Temporal Reasoning*, pages 241–350. Clarendon Press, Oxford, 1995.

[Benzen, 1959] S. Benzen. On the Topology of the Genetic Fine Structure. *Proceedings of the National Academy of Science*, 45(10):1607–1620, 1959.

[Berleant and Kuipers, 1997] D. Berleant and B. Kuipers. Qualitative and Quantitative Simulation: Bridging the Gap. *Artificial Intelligence*, 95(2):215–255, 1997.

[Berman, 1980] L. Berman. The Complexity of Logical Theories. *Theoretical Computer Science*, 11:71–78, 1980.

[Bernholtz, 1995] O. Bernholtz. *Model Checking for Branching Time Temporal Logics*. PhD thesis, The Technion, Israel, 1995.

[Bertoli *et al.*, 2001] P. Bertoli, A. Cimatti, M. Roveri, and P. Traverso. Planning in Non-Deterministic Domains under Partial Observability via Symbolic Model-Checking. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 2001.

[Berzuini *et al.*, 1989] C. Berzuini, R. Bellazzi, and S. Quaglini. Temporal Reasoning with Probabilities. In *Proceedingsof International Conference on Uncertainty in Artificial Intelligence (UAI)*, 1989.

[Berzuini *et al.*, 1997] C. Berzuini, N.G. Best, W.R. Gilks, and C. Larizza. Dynamic Conditional Independence Models and Markov Chain Monte Carlo Methods. *Journal of the American Statistical Association*, 92:1403–1412, 1997.

[Bessière *et al.*, 1996] C. Bessière, A. Isli, and G. Ligozat. Global Consistency in Interval Algebra Networks: Tractable Subclasses. In *Proceedings of the Fifteenth European Conference on Artificial Intelligence (ECAI)*, 1996.

[Bessière, 1996] C. Bessière. A Simple Way to Improve Path-Consistency in Interval Algebra Networks. In *Proceedings of the Thirteenth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 375–380, Portland, OR, 1996.

[Bessière, 1997] C. Bessière. Personal communication, August 1997.

[Beth, 1955] E. Beth. Semantic Entailment and Formal Derivability. *Mededelingen der Koninklijke Nederlandse Akad. van Wetensch*, 18, 1955.

[Bettini *et al.*, 1996] C. Bettini, X. S. Wang, and S. Jajodia. Testing Complex Temporal Relationships Involving Multiple Granularities and its Application to Data Mining. In *Proceedings of International Conference on Principles of Database Systems (PODS)*, pages 68–78, 1996.

[Bettini *et al.*, 1998a] C. Bettini, C. Dyreson, W. Evans, R. Snodgrass, and X. S. Wang. A Glossary of Time Granularity Concepts. In O. Etzion, S. Jajodia, and S. M. Sripada, editors, *Temporal Databases: Research and Practice*, volume 1399 of *Lecture Notes in Computer Science*, pages 406–413. Springer-Verlag, 1998.

[Bettini *et al.*, 1998b] C. Bettini, X. S. Wang, and S. Jajodia. An Architecture for Supporting Interoperability among Temporal Databases. In O. Etzion, S. Jajodia, and S. M. Sripada, editors, *Temporal Databases: Research and Practice*, volume 1399 of *Lecture Notes in Computer Science*, pages 26–55. Springer-Verlag, 1998.

[Bettini *et al.*, 1998c] C. Bettini, X. S. Wang, and S. Jajodia. A General Framework for Time Granularity and its Application to Temporal Reasoning. *Annals of Mathematics and Artificial Intelligence*, 1(22):29–58, 1998.

[Bettini *et al.*, 1998d] C. Bettini, X. S. Wang, S. Jajodia, and J-L. Lin. Discovering Frequent Event Patterns with Multiple Granularities in Time Sequences. *IEEE Transactions on Knowledge and Data Engineering*, 2(10):222–237, 1998.

[Bettini *et al.*, 1998e] C. Bettini, X.S. Wang, E. Bertino, and S. Jajodia. Semantic Assumptions and Query Evaluation in Temporal Databases. *IEEE Transactions on Knowledge and Data Engineering*, 10(2):277–296, 1998.

[Bettini *et al.*, 2000] C. Bettini, S. Jajodia, and X. S. Wang. *Time Granularities in Databases, Data Mining, and Temporal Reasoning*. Springer, Berlin, Germany, 2000.

[Bettini *et al.*, 2003] C. Bettini, S. Mascetti, and V. Pupillo. GSTP: A Temporal Reasoning System Supporting Multi-Granularity Temporal Constraints. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI); Intelligent Systems Demonstrations*, pages 1633–1634. Morgan Kaufmann, 2003.

[Bettini, 1997] C. Bettini. Time Dependent Concepts: Representation and Reasoning Using Temporal Description Logics. *Data and Knowledge Engineering*, 22(1):1–38, 1997.

[Bidoit *et al.*, 2004] N. Bidoit, S. de Amo, and L. Segoufin. Order Independent Temporal Properties. *Journal of Logic and Computation*, 2004.

[Birman, 1991] K. P. Birman. The Process Group Approach to Reliable Distributed Computing. Techanical Report TR91-1216, Department of Computer Science, Cornell University, July 1991.

[Bitner and Reingold, 1975] J.R. Bitner and E.M. Reingold. Backtrack Programming Techniques. *Journal of the ACM*, 18:651–655, 1975.

[Bittner and Steel, 1998] T. Bittner and J. Steel. A Boundary Sensitive Approach to Qualitative Location. *Annals of Mathematics and Artificial Intelligence*, 24(1-2):93–114, 1998.

[Bittner, 2002] T. Bittner. Approximate Qualitative Temporal Reasoning. *Annals of Mathematics and Artificial Intelligence*, 36(1-2):39–80, 2002.

[Bjorner *et al.*, 1995] N. Bjorner, A. Browne, E. Chang, M. Colón, A. Kapur, Z. Manna, H. B. Sipma, and T. E. Uribe. *STeP: The Stanford Temporal Prover Educational Release Version 1.0 User's Manual*. Computer Science Department, Stanford University, California 94305, November 1995.

[Blackburn *et al.*, 2001] P. Blackburn, M. de Rijke, and Yde Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, Cambridge, England, 2001.

[Blum and Furst, 1995] A. Blum and M. Furst. Fast Planning through Plan-graph Analysis. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1995.

[Blum, 1982] R.L. Blum. Discovery and Representation of Causal Relationships from a Large Time-Oriented Clinical Database: The RX Project. In D.A. Lindberg and P.L. Reichartz, editors, *Lecture Notes in Medical Informatics*, volume 19. Springer-Verlag, New York, 1982.

[Blumsohn, 1991] G. Blumsohn. *Three Essays on Law and Information in the Law of Damages*. Dissertation Information Service. UMI, 1991.

[Blythe, 1995] J. Blythe. AI Planning in Dynamic, Uncertain Domains. In *Proceedings of AAAI Spring Symposium on Extending Theories of Action*, 1995.

[Blythe, 1999] J. Blythe. An Overview of Planning Under Uncertainty. *Lecture Notes in Computer Science*, 1600:85–110, 1999.

[Boaz and Shahar, 2003] D. Boaz and Y. Shahar. Idan: A Distributed Temporal-Abstraction Mediator for Medical Databases. In *Proceedings of the Ninth Conference on Artificial Intelligence in Medicine — Europe (AIME)*, Protaras, Cyprus, 2003.

[Boddy and Dean, 1994] M. Boddy and T. Dean. Deliberation Scheduling for Problem Solving in Time-Constrained Enviroments. *Artificial Intelligence*, 67(2):245–285, 1994.

[Boddy, 1993] Mark Boddy. Temporal Reasoning for Planning and Scheduling. *SIGART Bulletin*, 4(3):17–20, 1993.

[Böhlen *et al.*, 1996a] M. Böhlen, J. Chomicki, R.T. Snodgrass, and D. Toman. Querying TSQL2 Databases with Temporal Logic. In *International Conference on Extending Database Technology (EDBT)*, Avignon, France, 1996. Springer Verlag, LNCS 1057.

[Böhlen *et al.*, 1996b] M. Böhlen, R.T. Snodgrass, and M.D. Soo. Coalescing in Temporal Databases. In *International Conference on Very Large Data Bases (VLDB)*, pages 180–191, 1996.

[Bolotov and Fisher, 1997] A. Bolotov and M. Fisher. A Resolution Method for CTL Branching-Time Temporal Logic. In *Proceedings of the Fourth International Workshop on Temporal Representation and Reasoning (TIME)*, Daytona Beach, Florida, May 1997. IEEE Computer Society Press.

[Bonet and Geffner, 1997] B. Bonet and H. Geffner. Planning as Heuristic Search: New Results. In *Proceedings of the Fourth European Conference on Planning (ECP)*. Springer-Verlag, 1997.

[Bonet and Geffner, 2000] B. Bonet and H. Geffner. Planning with Incomplete Information as Heuristic Search in Belief Space. In *Proceedings of the Fifth International Conference on AI Planning and Scheduling (AIPS)*, 2000.

[Bonet *et al.*, 1997] B. Bonet, G. Loerincs, and H. Geffner. A Robust and Fast Action Selection Mechanism for Planning. In *Proceedings of the Fourteenth National Conference on AI (AAAI)*, pages 714–719. AAAI/MIT Press, 1997.

[Bordini *et al.*, 2003a] R. Bordini, M. Fisher, C. Pardavila, and M. Wooldridge. Model Checking AgentSpeak. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), Melbourne, Australia, 14–18 July*, 2003.

[Bordini *et al.*, 2003b] R. Bordini, M. Fisher, W. Visser, and M. Wooldridge. Verifiable Multi-Agent Programs. In *Proceedings of the First International Workshop on Programming Multiagent Systems: languages, frameworks, techniques and tools (PROMAS), Melbourne, Australia*, 2003.

[Bordini *et al.*, 2003c] R. Bordini, W. Visser, M. Fisher, C. Pardavila, and M. Wooldridge. Model Checking Multi-Agent Programs with CASP. In *Proceedings of the Fifteenth Conference on Computer-Aided Verification (CAV), Boulder, USA, 8–12 July*, 2003.

[Borg *et al.*, 1983] A. Borg, J. Baumbach, and S. Glazer. A Message System Supporting Fault Tolerance. In *Proceedings of the Ninth ACM Symposium on Operating System Principles*, pages 90–99, New Hampshire, October 1983. ACM. (In ACM Operating Systems Review, vol. 17, no. 5).

[Boutilier and Goldszmidt, 1996] C. Boutilier and M. Goldszmidt. The Frame Problem and Bayesian Network Action Representations. In *Proceedings of the Eleventh Biennial Canadian Conference on Artificial Intelligence*, May 1996.

[Boutilier *et al.*, 1995a] C. Boutilier, T. Dean, and S. Hanks. Planning Under Uncertainty: Structural Assumptions and Computational Leverage. In *Proceedings of the Second European Planning Workshop*, 1995.

[Boutilier *et al.*, 1995b] C. Boutilier, R. Dearden, and M. Goldszmidt. Exploiting Structure in Policy Construction. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1995.

[Boutilier *et al.*, 1999] C. Boutilier, T. Dean, and S. Hanks. Decision Theoretic Planning: Structural Assumptions and Computational Leverage. *Journal of AI Research*, 11:1–94, 1999.

[Boyen and Koller, 1998] X. Boyen and D. Koller. Approximate Learning of Dynamic Models. In *Proceedings of NIPS*, pages 396–402, 1998.

[Bradshaw *et al.*, 1999] J. Bradshaw, M. Greaves, H. Holmback, T. Karygiannis, B. Silverman, N. Suri, and A. Wong. Agents for the Masses? *IEEE Intelligent Systems*, 14(2), 1999.

[Brajnik and Clancy, 1998] G. Brajnik and D. J. Clancy. Focusing Qualitative Simulation Using Temporal Logic: Theoretical Foundations. *Annals of Mathematics and Artificial Intelligence*, 22(1-2):59–86, 1998.

[Brass *et al.*, 1998] S. Brass, J. Dix, I. Niemelä, and T. Przymusinski. A Comparison of the Static and the Disjunctive Well-Founded Semantics and its Implementation. In *Proceedings of International Conference on Knowledge Representation and Reasoning (KR)*, pages 74–85, 1998.

[Bratman, 1990] M. E. Bratman. What is intention? In P. R. Cohen, J. L. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 15–32. MIT Press, 1990.

[Bresina *et al.*, 2002] J. Bresina, R. Dearden, N. Meuleau, D. Smith, and R. Washington. Planning under Continuous Time and Resource Uncertainty: A Challenge for AI. In *Proceedings of AIPS Workshop on Temporal Planning*, 2002.

[Bresolin *et al.*, 2004] D. Bresolin, A. Montanari, and G. Puppis. Time Granularities and Ultimately Periodic Automata. In *Ninth European Conference on Logics in Artificial Intelligence (JELIA)*, volume 3229 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 2004.

[Broy, 1997] M. Broy. Refinement of Time. *Lecture Notes in Computer Science*, 1231:44–63, 1997.

[Bruce, 1972] B. Bruce. A Model for Temporal References and its application in a Question Answering Program. *Artificial Intelligence*, 4:1–25, 1972.

[Bruneel and Clarebout, 1994] G. Bruneel and P. Clarebout. Een implementatie van SLD-NFA+CLP(R) en zijn toepassing voor het redeneren over continue verandering. Masterthesis, supervisors: M. Denecker and D. De Schreye, Dept. of Computing, K.U.Leuven, in Dutch, 1994.

[Brusoni *et al.*, 1994] V. Brusoni, L. Console, B. Pernici, and P. Terenziani. LaTeR: A General Purpose Manager of Temporal Information. In *Proceedings of the Eighth International Symposium on Methodologies for Intelligent Systems (ISMIS)*, volume 869 of *Lecture Notes in Computer Science*. Springer-Verlag, 1994.

[Brusoni *et al.*, 1995a] V. Brusoni, L. Console, B. Pernici, and P. Terenziani. Extending Temporal Relational Databases to Deal with Imprecise and Qualitative Temporal Information. In J. Clifford and A. Tuzhilin, editors, *Recent Advances in Temporal Databases (Proceedings of the International Workshop on Temporal Databases, Zürich, Switzerland, September 1995)*, Workshops in Computing. Springer, 1995.

[Brusoni *et al.*, 1995b] V. Brusoni, L. Console, and P. Terenziani. On the Computational Complexity of Querying Bounds on Differences Constraints. *Artificial Intelligence*, 74(2):367–379, 1995.

[Brusoni *et al.*, 1997] V. Brusoni, L. Console, B. Pernici, and P. Terenziani. LaTeR: An Efficient, General Purpose Manager of Temporal Information. *IEEE Expert*, 12(4):56–64, August 1997.

[Brusoni *et al.*, 1998] V. Brusoni, L. Console, P. Terenziani, and D. Theseider Dupré. A spectrum of definitions for temporal model-based diagnosis. *Artificial Intelligence*, 102(1):39–80, 1998.

[Brusoni *et al.*, 1999] V. Brusoni, L. Console, P. Terenziani, and B. Pernici. Qualitative and Quantitative Temporal Constraints and Relational Databases: Theory, Architecture, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 1(6):948–968, 1999.

[Bruss and Meyer, 1980] A.R. Bruss and A.R. Meyer. On Time-Space Classes and their Relation to the Theory of Real Addition. *Theoretical Computer Science*, 11:59–69, 1980.

[Bruynooghe, 1981] M. Bruynooghe. Solving Combinatorial Search Problems by Intelligent Backtracking. *Information Processing Letters*, 12(1), 1981.

[Brzoska, 1991] Ch. Brzoska. Temporal Logic Programming and its Relation to Constraint Logic Programming. In *Proceedings of International Logic Programming Symposium (ILPS)*, 1991.

[Brzoska, 1993] Ch. Brzoska. Temporal Logic Programming with Bounded Universal Goals. In *Proceedings of the Tenth International Conference on Logic Programming (ICLP)*, pages 239–256, 1993.

[Brzoska, 1995] Ch. Brzoska. Temporal Logic Programming in Dense Time. In *Proceedings of International Logic Programming Symposium (ILPS)*, 1995.

[Brzozowski and Leiss, 1980] J. Brzozowski and E. Leiss. Finite Automata, and Sequential Networks. *Theoretical Computer Science*, 10, 1980.

[Buchanan and Shortliffe, 1984] B.G. Buchanan and E.H. Shortliffe, editors. *Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, USA, 1984.

[Büchi, 1960] J. R. Büchi. Weak Second-Order Arithmetic and Finite Automata. *Z. Math. Logik Grundlag. Math.*, 6:66–92, 1960.

[Büchi, 1962] J.R. Büchi. On a Decision Method in Restricted Second Order Arithmetic. In *Logic, Methodology, and Philosophy of Science: Proceedings of 1960 International Congress*, pages 1–11. Stanford University Press, 1962.

[Bulygin, 1982] E. Bulygin. Time and Validity. In A. Martino, editor, *Deontic Logic, Computational Liguistics and Information Systems*, chapter Vol. II, pages 65–81. North-Holland, 1982.

[Burgess, 1982] J. P. Burgess. Axioms for Tense Logic I: ‘Since’ and ‘Until’. *Notre Dame J. Formal Logic*, 23(2):367–374, 1982.

[Burgess, 1984] J.P. Burgess. Basic Tense Logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic, volume II*, pages 89–133. D. Reidel Publishing Company, 1984.

[Bylander, 1994] T. Bylander. The Computational Complexity of Propositional STRIPS Planning. *Artificial Intelligence*, 69(1-2), 1994.

[Caironi *et al.*, 1997] P. Caironi, L. Portoni, C. Combi, F. Pinciroli, and S Ceri. HyperCare: A Prototype of an Active Database for Compliance with Essential Hypertension Therapy Guidelines. In *Proceedings of the AMIA Annual Fall Symposium (formerly the Symposium on Computer Applications in Medical Care)*, pages 288–292, Philadelphia, USA, 1997. Hanley & Belfus.

[Calvanese *et al.*, 2001] D. Calvanese, G. De Giacomo, M. Lenzerini, and D. Nardi. Reasoning in Expressive Description Logics. In *Handbook of Automated Reasoning*, pages 1581–1634. Elsevier, 2001.

[Carlson and Pelletier, 1995] G. Carlson and J. Pelletier, editors. *The Generic Book*. University of Chicago Press, Chicago, 1995.

[Carton and Thomas, 2002] O. Carton and W. Thomas. The Monadic Theory of Morphic Infinite Words and Generalizations. *Information and Computation*, 176:51–65, 2002.

[Casati and Varzi, 1996] R. Casati and A. Varzi, editors. *Events*, volume 15 of *The International Research Library of Philosophy*. Dartmouth Publishing, Aldershot, 1996.

[Castellini *et al.*, 2001] C. Castellini, E. Giunchiglia, and A. Tacchella. C-Plan: a Conformant Planner based on Satisfiability. In *Proceedings of International Joint Conference on Artificial Intelligence*, 2001.

[Cavalli and Fariñas del Cerro, 1984] A. Cavalli and L. Fariñas del Cerro. A Decision Method for Linear Temporal Logic. In R. E. Shostak, editor, *Proceedings of the Seventh International Conference on Automated Deduction (CADE)*, volume 170 of *Lecture Notes in Computer Science*, pages 113–127. Springer-Verlag, 1984.

[Cervoni *et al.*, 1994] R. Cervoni, A. Cesta, and A. Oddi. Managing Dynamic Temporal Constraint Networks. In *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS)*, pages 196–201, Chicago, USA, 1994. AAAI press.

[Cesta and Oddi, 1995] A. Cesta and A. Oddi. A Formal Domain Description Language for a Temporal Planner. In *Topics in AI: Proceedings of the Fourth Congress of the Italian Association for AI (AI*AI)*, pages 255–260, 1995.

[Cesta and Oddi, 1996] A. Cesta and A. Oddi. Gaining Efficiency and Flexibility in the Simple Temporal Problem. In L. Chittaro, S.D. Goodwin, H.J. Hamilton, and A. Montanari, editors, *Proceedings of the Third International Workshop on Temporal Representation and Reasoning (TIME)*, 1996.

[Chakravarty and Shahar, 2000] S. Chakravarty and Y. Shahar. A Constraint-Based Specification of Periodic Patterns in Time-Oriented Data. *Annals of Mathematics and Artificial Intelligence*, 30:1–4, 2000.

[Chakravarty and Shahar, 2001] S. Chakravarty and Y. Shahar. Specification and Detection of Periodicity in Clinical Data. *Methods of Information in Medicine*, 40(5):410–420, 2001. (Reprinted in: Haux, R., and Kulikowski, C. (eds), *Yearbook of Medical Informatics 2003*, Stuttgart: F.K. Schattauer and The International Medical Informatics Association, pp.296-306).

[Chandra *et al.*, 1981a] A. Chandra, J.Y. Halpern, A. Meyer, and R. Parikh. Equations between Regular Terms and an Application to Process Logic. In *Proceedings of the Thirteenth Annual ACM Symposium on Theory of Computation (STOC)*, pages 384–390, Milwaukee, USA, 1981.

[Chandra *et al.*, 1981b] A. Chandra, D. Kozen, and L. Stockmeyer. Alternation. *Journal of the ACM*, 28:114–133, 1981.

[Chaochen and Hansen, 1998] Z. Chaochen and M. Hansen. An Adequate First Order Interval Logic. In W.P. de Roever, H. Langmaak, and A. Pnueli, editors, *Compositionality: the Significant Difference*, volume 1536 of *LNCS*, pages 584–608. Springer, 1998.

[Chapman, 1987] D. Chapman. Planning for Conjunctive Goals. *Artificial Intelligence*, 32(3):333–377, 1987.

[Charniak and McDermott, 1985] E. Charniak and D. McDermott. *Introduction to Artifical Intelligence*. Addison-Wesley, 1985.

[Charniak, 1991] E. Charniak. Bayesian Networks without Tears. *AI Magazine*, 12(4):50–63, December 1991.

[Cheeseman *et al.*, 1991] P. Cheeseman, B. Kanefsky, and W.M. Taylor. Where the *really* hard problems are. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 331–337. Morgan Kaufmann, 1991.

[Chemilieu-Gendrau, 1987] M. Chemilieu-Gendrau. *Le Role du Temps dans la Formation du Droit International*. Droit International. Editions Pedone, 1987.

[Chen and Warren, 1989] W. Chen and D. Warren. C-logic of Complex Objects. In *Proceedings of the Eighth ACM-SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, 1989.

[Chen and Zaniolo, 1999] C. X. Chen and C. Zaniolo. Universal Temporal Extensions for Database Languages. In *Proceedings of IEEE International Conference on Data Engineering*, pages 428–437, 1999.

[Chen *et al.*, 1995] W. Chen, T. Swift, and D. Warren. Efficient Top-Down Computation of Queries under the Well-Founded Semantics. *Journal of Logic Programming*, 24(3):161–201, 1995.

[Chen *et al.*, 2000] J. Chen, D. J. DeWitt, F. Tian, and Y. Wang. NiagaraCQ: A Scalable Continuous Query System for Internet Databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 379–390, 2000.

[Chien *et al.*, 2001] S-Y. Chien, V. J. Tsotras, and C. Zaniolo. Version Management of XML Documents. *SIGMOD Record*, 30(3):46–53, 2001.

[Chien *et al.*, 2002] S-Y. Chien, V. J. Tsotras, and C. Zaniolo. Efficient Schemes for Managing Multiversion XML Documents. *The VLDB Journal*, 11(4):332–353, 2002.

[Chimenti, 1990] D. Chimenti. The IDL System Prototype. *IEEE Transactions on Knowledge and Data Engineering*, 2(1):78–90, 1990.

[Chittaro and Dojat, 1997] L. Chittaro and M. Dojat. Using a General Theory of Time and Change in Patient Monitoring: Experiment and Evaluation. *Computers in Biology and Medicine*, 27(5):435–452, 1997.

[Chomicki and Imieliński, 1988] J. Chomicki and T. Imieliński. Temporal Deductive Databases and Infinite Objects. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 61–73, Austin, USA, March 1988.

[Chomicki and Niwinski, 1995] J. Chomicki and D. Niwinski. On the Feasibility of Checking Temporal Integrity Constraints. *Journal of Computer and System Sciences*, 51(3):523–535, December 1995.

[Chomicki and Toman, 1998] J. Chomicki and D. Toman. Temporal Logic in Information Systems. In J. Chomicki and G. Saake, editors, *Logics for Databases and Information Systems*, pages 31–70. Kluwer, 1998.

[Chomicki *et al.*, 1996] J. Chomicki, D. Goldin, and G. Kuper. Variable Independence and Aggregation Closure. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 40–48, Montréal, Canada, June 1996.

[Chomicki *et al.*, 2001] J. Chomicki, D. Toman, and M. H. Böhlen. Querying ATSQL Databases with Temporal Logic. *ACM Transactions on Database Systems*, 26(2):145–178, 2001.

[Chomicki *et al.*, 2003a] J. Chomicki, D. Goldin, G. Kuper, and D. Toman. Variable Independence in Constraint Databases. *IEEE Transactions on Knowledge and Data Engineering*, 15(6):1422–1436, 2003.

[Chomicki *et al.*, 2003b] J. Chomicki, G. Saake, and R. van der Meyden, editors. *Logics for Emerging Applications of Databases*. Springer, 2003.

[Chomicki, 1994] J. Chomicki. Temporal Query Languages: A Survey. In Gabbay, D. and Ohlbach, H., editor, *First International Conference on Temporal Logic (TIME)*, volume 827 of *Lecture Notes in Artificial Intelligence*. Springer Verlag, 1994.

[Chomicki, 1995] J. Chomicki. Efficient Checking of Temporal Integrity Constraints Using Bounded History Encoding. *ACM Transactions on Database Systems*, 20(2):149–186, June 1995.

[Choueka, 1974] Y. Choueka. Theories of Automata on ω-Tapes: A Simplified Approach. *Journal of Computer and System Sciences*, 8:117–141, 1974.

[Ciapessoni *et al.*, 1993] E. Ciapessoni, E. Corsetti, A. Montanari, and P. San Pietro. Embedding Time Granularity in a Logical Specification Language for Synchronous Real-Time Systems. *Science of Computer Programming*, 20(1):141–171, 1993.

[Cimatti and Roveri, 1999] A. Cimatti and M. Roveri. Conformant Planning via Model Checking. In *Proceedings of European Conference on Planning (ECP)*, pages 21–34, 1999.

[Cimatti and Roveri, 2000] A. Cimatti and M. Roveri. Conformant Planning via Symbolic Model Checking. *Journal of AI Research*, 13:305–338, 2000.

[Cimatti *et al.*, 1998a] A. Cimatti, F. Giunchiglia, and R. W. Weyhrauch. A Many-Sorted Natural Deduction. *Computational Intelligence*, 14:134–149, 1998.

[Cimatti *et al.*, 1998b] A. Cimatti, M. Roveri, and P. Traverso. Automatic OBDD-Based Generation of Universal Plans in Non-Deterministic Domains. In *Proceedings of AAAI*, pages 875–881, 1998.

[Citrigno *et al.*, 1997] S. Citrigno, T. Eiter, W. Faber, G. Gottlob, C. Koch, N. Leone, C. Mateis, G. Pfeifer, and F. Scarcello. The DLV System: Model Generator and Application Front Ends. In *Proceedings of the Twelfth Workshop on Logic Programming*, pages 128–137, 1997.

[Clancey, 1985] W.J. Clancey. Heuristic Classification. *Artificial Intelligence*, 27(3):289–350, 1985.

[Clark, 1978] Keith L. Clark. Negation as Failure. In H. Gallaire and J. Minker, editors, *Logic and Databases*, pages 293–322. Plenum Press, 1978.

[Clarke and Emerson, 1981a] E. Clarke and E. Emerson. Synthesis of Synchronization Skeletons for Branching Time Temporal Logic. In *Proceedings of IBM Workshop on Logic of Programs*, pages 52–71, Yorktown Heights, USA, 1981. Springer, Berlin.

[Clarke and Emerson, 1981b] E.M. Clarke and E.A. Emerson. Design and Synthesis of Synchronization Skeletons using Branching Time Temporal Logic. In D. Kozen, editor, *Workshop on Logics of Programs*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71. Springer-Verlag, 1981.

[Clarke and Schlingloff, 2001] E.M. Clarke and B.-H. Schlingloff. Model Checking. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, pages 1635–1790. North-Holland, 2001.

[Clarke *et al.*, 1983] E.M. Clarke, E.A. Emerson, and A.P. Sistla. Automatic Verification of Finite State Concurrent Systems Using Temporal Logic Specifications: A Practical Approach. In *Proceedings of International Symposium on Principles of Programming Languages (POPL)*, pages 117–126. ACM Press, 1983.

[Clarke *et al.*, 1986] E.M. Clarke, E.A. Emerson, and A.P. Sistla. Automatic Verification of Finite State Concurrent Systems Using Temporal Logic Specifications. *ACM Transactions on Programming Languages and Systems*, 8(2):244–263, 1986.

[Clarke *et al.*, 1999] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. MIT Press, 1999.

[Clifford and Rao, 1988] J. Clifford and A. Rao. A Simple General Structure for Temporal Domains. In *Temporal Aspects of Information Systems*, pages 17–28. Elsevier, Amsterdam (NL), 1988.

[Clifford *et al.*, 1993] J. Clifford, A. Croker, and A. Tuzhilin. On the Completeness of Query Languages for Grouped and Ungrouped Historical Data Models. In Tansel, A. et. al., editor, *Temporal Databases: Theory, Design and Implementation*, chapter 20, pages 496–533. Benjamin/Cummings Pub. Co., 1993.

[Clifford *et al.*, 1994] J. Clifford, A. Croker, and A. Tuzhilin. On Completeness of Historical Relational Query Languages. *ACM Transactions on Database Systems*, 19(1):64–116, March 1994.

[Codd, 1971] E. F. Codd. Normalized Data Structure: A Brief Tutorial. In E. F. Codd and A. L. Dean, editors, *Proceedings of ACM-SIGFIDET Workshop on Data Description, Access and Control*, pages 1–17, San Diego, USA, 1971. ACM.

[Codd, 1972] E. F. Codd. Relational Completeness of Data Base Sub-Languages. In R. Rustin, editor, *Data Base Systems*, pages 33–64. Prentice-Hall, 1972.

[Cohen and Levesque, 1990] P. R. Cohen and H. J. Levesque. Intention is Choice with Commitment. *Artificial Intelligence*, 42:213–261, 1990.

[Cohen *et al.*, 1997] D. Cohen, P. Jeavons, and M. Koubarakis. Tractable disjunctive constraints. In *Proceedings of Constraint Programming (CP)*, volume 1330 of *Lecture Notes in Computer Science*, pages 478–490, Linz, Austria, 1997.

[Cohen *et al.*, 2000] D. Cohen, P. Jeavons, P. Jonsson, and M. Koubarakis. Building Tractable Disjunctive Constraints. *Journal of the ACM*, 47(5):826–853, 2000.

[Cohn, 1987] A.G. Cohn. A More Expressive Formulation of Many Sorted Logics. *Journal of Automated Reasoning*, 3(2):113–200, 1987.

[Combi *et al.*, 1994] C. Combi, F. Pinciroli, G. Musazzi, and C. Ponti. Managing and Displaying Different Time Granularities of Clinical Information. In J.G. Ozbolt, editor, *Proceedings of Eighteenth Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pages 954–958, Philadelphia, USA, 1994. Hanley & Belfus.

[Combi *et al.*, 1995] C. Combi, F. Pinciroli, M. Cavallaro, and G. Cucchi. Querying Temporal Clinical Databases with Different Time Granularities: the GCH-OSQL Language. In *Proceedings of the Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pages 326–330, New-Orleans, USA, 1995.

[Combi *et al.*, 2004] C. Combi, M. Franceschet, and A. Peron. Representing and Reasoning about Temporal Granularities. *Journal of Logic and Computation*, 14(1):51–77, 2004.

[Comrie, 1976] B. Comrie. *Aspect: an Introduction to the Study of Verbal Aspect and Related Problems*. Cambridge University Press, Cambridge, 1976.

[Condotta, 2000] J-F. Condotta. The Augmented Interval and Rectangle Networks. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 571–579, Breckenridge, CO, USA, April 2000. Morgan Kaufmann.

[Console and Terenziani, 1999] L. Console and P. Terenziani. Efficient Processing of Queries and Assertions about Qualitative and Quantitative Temporal Constraints. *Computational Intelligence*, 15(4):442–465, 1999.

[Console and Torasso, 1991a] L. Console and P. Torasso. A Spectrum of Logical Definitions of Model-Based Diagnosis. *Computational Intelligence*, 7:133–141, 1991.

[Console and Torasso, 1991b] L. Console and P. Torasso. On the Co–operation Between Abductive and Temporal Reasoning in Medical Diagnosis. *Artificial Intelligence in Medicine*, 3:291–311, 1991.

[Console *et al.*, 1991] L. Console, D. Theseider Dupre, and P. Torasso. On the Relationship Between Abduction and Deduction. *Journal of Logic and Computation*, 1(5):661–690, 1991.

[Cooper, 1990] G. Cooper. The Computational Complexity of Probabilistic Inference using Bayesian Belief Networks. *Artificial Intelligence*, 42, 1990.

[Cormen *et al.*, 1990] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[Corsetti *et al.*, 1991a] E. Corsetti, E. Crivelli, D. Mandrioli, A. Montanari, A. Morzenti, P. San Pietro, and E. Ratto. Dealing with Different Time Scales in Formal Specifications. In *International Workshop on Software Specification and Design*, pages 92–101, 1991.

[Corsetti *et al.*, 1991b] E. Corsetti, A. Montanari, and E. Ratto. Dealing with Different Time Granularities in Formal Specifications of Real-Time Systems. *Journal of Real-Time Systems*, 3(2):191–215, 1991.

[Cousins and Kahn, 1991] S.B. Cousins and M.G. Kahn. The Visual Display of Temporal Information. *Artificial Intelligence in Medicine*, 3(6):341–357, 1991.

[Cowell *et al.*, 1999] R.C. Cowell, A.P. Dawid, S.L. Lauritzen, and D.J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer-Verlag, NewYork, 1999.

[Culik II *et al.*, 1984] K. Culik II, A. Salomaa, and D. Wood. Systolic Tree Acceptors. *R.A.I.R.O Informatique Théorique*, 18:53–69, 1984.

[D, 1982] McDermott D. A Temporal Logic for Reasoning about Processes and Plans. *Cognitive Science*, 6:101–155, 1982.

[Dagum and Galper, 1993] P. Dagum and A. Galper. Forecasting Sleep Apnea with Dynamic Network Models. In *Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 64–71, 1993.

[Dal Lago and Montanari, 2001] U. Dal Lago and A. Montanari. Calendars, Time Granularities, and Automata. In *Proceedings of the International Symposium on Spatial and Temporal Databases*, volume 2121 of *Lectures Notes on Computer Science*, pages 279–298, Los Angeles, USA, 2001.

[Dal Lago *et al.*, 2003a] U. Dal Lago, A. Montanari, and G. Puppis. Time Granularities, Calendar Algebra, and Automata. Technical Report 4, Dipartimento di Matematica e Informatica, Università di Udine, Italy, February 2003.

[Dal Lago *et al.*, 2003b] U. Dal Lago, A. Montanari, and G. Puppis. Towards Compact and Tractable Automaton-based Representations of Time Granularity. In *Proceedings of the Eighth Italian Conference on Theoretical Computer Science (ICTCS)*, volume 2841 of *LNCS*, pages 72–85. Springer, October 2003.

[Das and Musen, 1994] A.K. Das and M.A. Musen. A Temporal Query System for Protocol–Directed Decision Support. *Methods of Information in Medicine*, 33:358–370, 1994.

[Date *et al.*, 2003] C. J. Date, H. Darwen, and N. A. Lorentzos. *Temporal Data and the Relational Model*. Morgan Kaufman, 2003.

[Davidson, 1967] D. Davidson. The Logical Form of Action Sentences. In N. Rescher, editor, *The Logic of Decision and Action*. University of Pittsburgh Press, 1967.

[Davidson, 1969] D. Davidson. The Individuation of Events. In N. Rescher, editor, *Essays in Honor of Carl G. Hempel*. D. Reidel, Dordrecht, 1969. (Reprinted in D. Davidson, *Essays on Actions and Events*, Oxford, Clarendon Press, 1980, pages 163-180).

[Davis, 1992] E. Davis. Infinite Loops in Finite Time: Some Observations. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 47–58. Morgan-Kaufmann, 1992.

[Dawid, 1992] A. P. Dawid. Applications of a General Propagation Algorithm for Probabilistic Expert Systems. *Statistics and Computing*, 2:25–36, 1992.

[de Zegher-Geets *et al.*, 1988] I.M. de Zegher-Geets, A.G. Freeman, M.G. Walker, R.L. Blum, and G. Wiederhold. Summarization and Display of On-Line Medical Records. *M.D. Computing*, 5:38–46, 1988.

[Dean and Boddy, 1988] T. Dean and M. Boddy. Reasoning About Partially Ordered Events. *Artificial Intelligence*, 36:375–399, 1988.

[Dean and Kanazawa, 1989] T. Dean and K. Kanazawa. A Model for Reasoning about Persistence and Causation. *Computational Intelligence*, 5:142–150, 1989.

[Dean and McDermott, 1987] T. Dean and D.V. McDermott. Temporal Data Base Management. *Artificial Intelligence*, 32:1–55, 1987.

[Dean, 1989] T. Dean. Using Temporal Hierarchies to Efficiently Maintain Large Temporal Databases. *Journal of ACM*, 36(4):687–718, 1989.

[Dechter *et al.*, 1989] R. Dechter, I. Meiri, and J. Pearl. Temporal Constraint Networks. In R. Brachman, H. Levesque, and R. Reiter, editors, *Proceedings of First International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 83–93, Toronto, Canada, 1989.

[Dechter *et al.*, 1991] R. Dechter, I. Meiri, and J. Pearl. Temporal Constraint Networks. *Artificial Intelligence*, 49(1-3):61–95, 1991. Special Volume on Knowledge Representation.

[Dechter, 1992] R. Dechter. From Local to Global Consistency. *Artificial Intelligence*, 55:87–107, 1992.

[Degtyarev *et al.*, 2003] A. Degtyarev, M. Fisher, and B. Konev. Monodic Temporal Resolution. In *Proceedings of Conference on Automated Deduction (CADE)*, Lecture Notes in Computer Science. Springer, July 2003.

[Delgrande and Gupta, 1996] J.P. Delgrande and A. Gupta. A Representation for Efficient Temporal Reasoning. In *Proceedings of the Thirteenth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 381–388, Portland, OR, 1996.

[Delgrande and Gupta, 2002] J.P. Delgrande and A. Gupta. Updating $<=,<$-Chains. *Information Processing Letters*, 83(5):261–268, 2002.

[Delgrande *et al.*, 1999] J.P. Delgrande, A. Gupta, and T. Van Allen. Point Based Approaches to Qualitative Temporal Reasoning. In *Proceedings of AAAI Conference*, pages 739–744, 1999.

[Delgrande *et al.*, 2001] J. Delgrande, A. Gupta, and T. Van Allen. A Comparison of Point-based Approaches to Qualitative Temporal Reasoning. *Artificial Intelligence*, 131:135–170, 2001.

[Dembinski and Maluszynski, 1985] P. Dembinski and J. Maluszynski. AND-Parallelism with Intelligent Backtracking for Annotated Logic Programs. In V. Saraswat and K. Ueda, editors, *Proceedings of the International Symposium on Logic Programming (ILPS)*, pages 25–38, 1985.

[Demri, 2004] S. Demri. LTL over Integer Periodicity Constraints (Extended Abstract). In *Proceedings of the Seventh International Conference on Foundations of Software Science and Computation Structures (FOSSACS)*, volume 2987 of *Lecture Notes in Computer Science*, pages 121–135, Heidelberg, Germany, 2004. Springer-Verlag.

[Denecker and De Schreye, 1992] M. Denecker and D. De Schreye. SLDNFA: an Abductive Procedure for Normal Abductive Programs. In K.R. Apt, editor, *Proceedings of the International Joint Conference and Symposium on Logic Programming*, pages 686–700. MIT Press, 1992.

[Denecker and De Schreye, 1993] M. Denecker and D. De Schreye. Representing incomplete Knowledge in Abductive Logic Programming. In *Proceedings of International Logic Programming Symposium (ILPS)*, pages 147–164, Vancouver, Canada, 1993.

[Denecker and De Schreye, 1998] M. Denecker and D. De Schreye. SLDNFA: an Abductive Procedure for Abductive Logic Programs. *Journal of Logic Programming*, 34(2):111–167, 1998.

[Denecker *et al.*, 1992] M. Denecker, L. Missiaen, and M. Bruynooghe. Temporal reasoning with Abductive Event Calculus. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*. John Wiley and sons, 1992.

[Denecker *et al.*, 1996] M. Denecker, K. Van Belleghem, G. Duchatelet, F. Piessens, and D. De Schreye. A Realistic Experiment in Knowledge Representation Using Open Event Calculus: Protocol Specification. In M. Maher, editor, *Proceedings of the International Joint Conference and Symposium on Logic Programming*, pages 170–184. MIT Press, 1996.

[Denecker, 1993] M. Denecker. *Knowledge Representation and Reasoning in Incomplete Logic Programming*. PhD thesis, Department of Computer Science, K.U.Leuven, 1993.

[Denecker, 1995] M. Denecker. A Terminological Interpretation of (Abductive) Logic Programming. In V.W. Marek, A. Nerode, and M. Truszczynski, editors, *International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, volume 928 of *Lecture Notes in Artificial Intelligence*, pages 15–29. Springer, 1995.

[Dettori and Puppo, 1996] G. Dettori and E. Puppo. How Generalization Interacts with the Topological and Metric Structure of Maps. In *Proceedings of International Symposium on Spatial Data Handling*, pages 27–38, 1996.

[Devanbu and Litman, 1996] P. T. Devanbu and D. J. Litman. Taxonomic Plan Reasoning. *Artificial Intelligence*, 84:1–35, 1996.

[Devogele *et al.*, 1996] T. Devogele, J. Trevisan, and L. Raynal. Building a Multi-Scale Database with Scale-Transition Relationship. In *Proceedings of International Symposium on Spatial Data Handling*, 1996.

[Dexter *et al.*, 2001] P.R. Dexter, S. Perkins, M.J. Overhage, K. Maharry, R.B. Kohler, and C.J. McDonald. A Computerized Reminder System to Increase the use of Preventive Care for Hospitalized Patients. *New England Journal of Medicine*, 345(13):965–970, 2001.

[Dimopoulos *et al.*, 1997] Y. Dimopoulos, B. Nebel, and J. Koehler. Encoding Planning Problems in Non-Monotonic Logic Programs. In *Proceedings of European Conference on Planning (ECP)*, pages 169–181, 1997.

[Dix, 1991] J. Dix. Classifying Semantics of Logic Programs. In *Proceedings of International Workshop in Logic Programming and Nonmonotonic Reasoning (LPNMR)*, pages 166–180, Washington D.C., USA, 1991.

[Dixon *et al.*, 1998] C. Dixon, M. Fisher, and M. Wooldridge. Resolution for Temporal Logics of Knowledge. *Journal of Logic and Computation*, 8(3):345–372, 1998.

[Dixon, 1996] C. Dixon. Search Strategies for Resolution in Temporal Logics. In M. A. McRobbie and J. K. Slaney, editors, *Proceedings of the Thirteenth International Conference on Automated Deduction (CADE)*, volume 1104 of *Lecture Notes in Artificial Intelligence*, pages 672–687, New Brunswick, USA, July/August 1996. Springer-Verlag.

[Dixon, 1998] C. Dixon. Temporal Resolution using a Breadth-First Search Algorithm. *Annals of Mathematics and Artificial Intelligence*, 22:87–115, 1998.

[Dixon, 2000] C. Dixon. Using Otter for Temporal Resolution. In *Advances in Temporal Logic*, volume 16 of *Applied Logic Series*, pages 149–166. Kluwer, 2000. (Proceedings the Second International Conference on Temporal Logic (ICTL).).

[Do and Kambhampati, 2001] M. Binh Do and S. Kambhampati. Sapa: a Domain-Independent Heuristic Metric Temporal Planner. In *Proceedings of European Conference on Planning (ECP)*, 2001.

[Doner, 1970] J. Doner. Tree Acceptors and Some of their Applications. *Journal of Computer and System Sciences*, 4:406–451, 1970.

[Donini *et al.*, 1996] F. M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in Description Logics. In G. Brewka, editor, *Principles of Knowledge Representation*, pages 191–236. CSLI Publications, Stanford, California, 1996.

[Doucet *et al.*, 2001] A. Doucet, N. DeFreitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, NewYork, 2001.

[Downes *et al.*, 1986] S.M. Downes, M.G. Walker, and R.L. Blum. Automated Summarization of On-line Medical Records. In R. Salamon, B. Blum, and M. Jorgensen, editors, *Proceedings of the Fifth Conference on Medical Informatics (MEDINFO)*, pages 800–804, Amsterdam, NL, 1986. North-Holland.

[Dowty, 1979] D. Dowty. *Word Meaning and Montague Grammar*. Kluwer Academic Publishers, Dordrecht, 1979.

[Drabble and Tate, 1994] B. Drabble and A. Tate. The Use of Optimistic and Pessimistic Resource Profiles to Inform Search in an Activity Based Planner. In *Proceedings of the Second Conference on AI Planning Systems (AIPS)*. AAAI Press, 1994.

[Drabble, 1993] B. Drabble. EXCALIBUR: A Program for Planning and Reasoning with Processes. *Artificial Intelligence*, 62(1):1–40, 1993.

[Drakengren and Jonsson, 1997a] T. Drakengren and P. Jonsson. Eight maximal tractable subclasses of Allen's algebra with metric time. *Journal of Artificial Intelligence Research*, 7:25–45, 1997.

[Drakengren and Jonsson, 1997b] T. Drakengren and P. Jonsson. Twenty-One Large Tractable Subclasses of Allen's Algebra. *Artificial Intelligence*, 93(1–2):297–319, 1997.

[Drakengren and Jonsson, 1998] T. Drakengren and P. Jonsson. A Complete Classification of Allen's Algebra Relative to Subsets Of Basic Relations. *Artificial Intelligence*, 106(2):205–219, 1998.

[Dung, 1993] P. Dung. Representing Actions in Logic Programming and its Application in Database Updates. In D. S. Warren, editor, *Proceedings of International Conference on Logic Programming (ICLP)*, pages 222–238, 1993.

[Durfee, 1988] E. H. Durfee. *Coordination of Distributed Problem Solvers*. Kluwer, 1988.

[Dvorak and Kuipers, 1989] D. Dvorak and B. Kuipers. Model-Based Monitoring of Dynamic Systems. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1238–1243, San Mateo, CA, 1989. Morgan Kaufmann.

[Dyreson and Snodgrass, 1994] C. Dyreson and R. Snodgrass. Temporal Granularity and Indeterminacy: Two Sides of the Same Coin. Technical Report 6, University of Arizona, Tucson, USA, 1994.

[Edelkamp and Helmert, 2000] S. Edelkamp and M. Helmert. On the Implementation of MIPS. In *Proceedings of Workshop on Decision-Theoretic Planning, Artificial Intelligence Planning and Scheduling (AIPS)*, pages 18–25. AAAI-Press, 2000.

[Egenhofer and Franzosa, 1991] M. Egenhofer and R. Franzosa. Point Set Topological Spatial Relations. *International Journal of Geographical Information Systems*, 5(2):161–174, 1991.

[Eiter *et al.*, 2000a] T. Eiter, W. Faber, G. Gottlob, C. Koch, C. Mateis, N. Leone, G. Pfeifer, and F. Scarcello. The DLV System. In J. Minker, editor, *Pre-prints of Workshop on Logic-Based AI*, 2000.

[Eiter *et al.*, 2000b] T. Eiter, W. Faber, N. Leone, and G. Pfeifer. Declarative problem solving in DLV. In J. Minker, editor, *Logic Based Artificial Intelligence*, pages 79–103. Kluwer Academic Publishers, 2000.

[Elgot and Rabin, 1966] C. Elgot and M. Rabin. Decidability and Undecidability of Second (First) Order Theory of Generalized Successor. *Journal of Symbolic Logic*, 31:169–181, 1966.

[Elgot, 1961] C. Elgot. Decision Problems for Finite Automata Design and Related Arithmetics. *Trans. Amer. Math. Soc.*, 98:21–52, 1961.

[Emerson and Clarke, 1982] E. Emerson and E. C. Clarke. Using Branching Time Temporal Logic to Synthesise Synchronisation Skeletons. *Science of Computer Programming*, 2, 1982.

[Emerson and Halpern, 1985] E. Emerson and J. Halpern. Decision Procedures and Expressiveness in the Temporal Logic of Branching Time. *Journal of Computer and System Sciences*, 30(1):1–24, 1985.

[Emerson and Halpern, 1986] E. Emerson and J. Halpern. 'Sometimes' and 'Not Never' Revisited: on Branching versus Linear Time. *Journal of the ACM*, 33, 1986.

[Emerson and Jutla, 1988] E. Emerson and C. Jutla. Complexity of Tree Automata and Modal Logics of Programs. In *Proceedings of the Twenty Ninth IEEE Conference on Foundations of Computer Science (FOCS)*. IEEE, 1988.

[Emerson and Lei, 1985] E. Emerson and C. Lei. Modalities for Model Checking: Branching Time Strikes Back. In *Proceedings of the Twelfth Symposium on Principles of Programming Languages (POPL)*, pages 84–96, 1985.

[Emerson and Sistla, 1984] E. Emerson and A. Sistla. Deciding Full Branching Time Logic. *Information and Control*, 61(3):175 – 201, 1984.

[Emerson, 1985] E. Emerson. Automata, Tableaux and Temporal Logics. In *Proceedings of the Workshop on Logics of Programs*, Brooklyn, USA, 1985.

[Emerson, 1990] E.A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science: Formal Models and Semantics*, chapter 16, pages 995–1072. Elsevier Science Publishers, 1990.

[Emerson, 1996] E. Emerson. Automated Temporal Reasoning for Reactive Systems. In F. Moller and G. Birtwistle, editors, *Logics for Concurrency*, pages 41–101. Springer Verlag, 1996.

[Enderton, 1972] H. B. Enderton. *A Mathematical Introduction To Logic*. Academic Press, 1972.

[Endriss, 2003] U. Endriss. *Modal Logics of Ordered Trees*. PhD thesis, Department of Computer Science, King's College London, 2003.

[Erdem and Lifschitz, 1999] E. Erdem and V. Lifschitz. Transformations of Logic Programs Related to Causality and Planning. In *Proceedings of Logic Programming and Non-Monotonic Reasoning (LPNMR)*, pages 107–116, 1999.

[Eshghi and Kowalski, 1989] K. Eshghi and R.A. Kowalski. Abduction Compared with Negation as Failure. In *Proceedings of the International Conference on Logic Programming (ICLP)*. MIT-Press, 1989.

[Eshghi, 1988a] K. Eshghi. Abductive Planning with Event Calculus. In R.A. Kowalski and K.A. Bowen, editors, *Proceedings of the Fifth International Conference and Symposium on Logic Programming*, pages 562–579, Cambridge, MA, 1988. MIT Press.

[Eshghi, 1988b] K. Eshghi. Abductive Planning with Event Calculus. In *Proceedings of International Conference on Logic Programming (ICLP)*, 1988.

[Euzenat, 1993] J. Euzenat. Représentation Granulaire du Temps. *Revue d'Intelligence Artificielle*, 7(3):329–361, 1993.

[Euzenat, 1995a] J. Euzenat. A Categorical Approach to Time Representation: First Study on Qualitative Aspects. In *Proceedings of the IJCAI Workshop on Spatial and Temporal Reasoning*, pages 145–152, 1995.

[Euzenat, 1995b] J. Euzenat. An Algebraic Approach to Granularity in Qualitative Space and Time Representation. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 894–900, 1995.

[Euzenat, 2001] J. Euzenat. Granularity in Relational Formalisms with Application to Time and Space Representation. *Computational Intelligence*, 17(4):703–737, 2001.

[Evans, 1989] C. Evans. Negation-as-Failure as an Approach to the Hanks and McDermott Problem. In *Proceedings of the Second International Symposium on Artificial Intelligence*, 1989.

[Fagan *et al.*, 1984] M. Fagan, J. C. Kunz, E. A. Feigenbaum, and J. J. Osborn. Extensions to the Rule-Based Formalism for a Monitoring Task. In Buchanan and Shortliffe [1984], pages 397–423.

[Fagin *et al.*, 1996] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning About Knowledge*. MIT Press, 1996.

[Ferrante and Geiser, 1977] J. Ferrante and J.R. Geiser. An Efficient Decision Procedure for the Theory of Rational Order. *Theoretical Computer Science*, 4(2):227–233, 1977.

[Ferrante and Rackoff, 1975] J. Ferrante and C. Rackoff. A Decision Procedure for the First Order Theory of Real Addition with Order. *SIAM Journal on Computing*, 4(1):69–76, 1975.

[Ferrante and Rackoff, 1979] J. Ferrante and C. Rackoff. *The Computational Complexity of Logical Theories*. Lecture Notes in Mathematics. Springer Verlag, 1979.

[Fiadeiro and Maibaum, 1994] J. L. Fiadeiro and T. Maibaum. Sometimes "tomorrow" is "sometime": Action Refinement in a Temporal Logic of Objects. *Lecture Notes in Computer Science*, 827:48–66, 1994.

[Fikes and Nilsson, 1971] R.E. Fikes and N.J. Nilsson. STRIPS: A New Approach to the Application of Theorem-Proving to Problem-Solving. *Artificial Intelligence*, 2(3):189–208, 1971.

[Finger and Gabbay, 1992] M. Finger and D.M. Gabbay. Adding a Temporal Dimension to a Logic System. *Journal of Logic, Language, and Information*, 1(3):203–233, 1992.

[Finger and Gabbay, 1996] M. Finger and D. Gabbay. Combining Temporal Logic Systems. *Notre Dame Journal of Formal Logic*, 37:204–232, 1996.

[Fischer and Rabin, 1974] M.J. Fischer and M.O. Rabin. Super-exponential Complexity of Presburger Arithmetic. In *Proceedings of the AMS Symposium on Complexity of Real Computational Processes*, volume III, 1974.

[Fisher and Ghidini, 1999] M. Fisher and C. Ghidini. Programming Resource-Bounded Deliberative Agents. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. Morgan Kaufmann, 1999.

[Fisher and Ghidini, 2002] M. Fisher and C. Ghidini. The ABC of Rational Agent Programming. In *Proceedings of the First International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 849–856. ACM Press, July 2002.

[Fisher and Kakoudakis, 1999] M. Fisher and T. Kakoudakis. Flexible Agent Grouping in Executable Temporal Logic. In *Proceedings of Twelfth International Symposium on Languages for Intensional Programming (ISLIP)*. World Scientific Press, 1999.

[Fisher and Owens, 1992] M. Fisher and R. Owens. From the Past to the Future: Executing Temporal Logic Programs. In *Proceedings of Logic Programming and Automated Reasoning (LPAR)*, volume 624 of *Lecture Notes in Computer Science*, St. Petersberg, Russia, July 1992. Springer-Verlag.

[Fisher and Owens, 1995a] M. Fisher and R. Owens. An Introduction to Executable Modal and Temporal Logics. In *Executable Modal and Temporal Logics*, volume 897 of *Lecture Notes in Artificial Intelligence*, pages 1–20, Heidelberg, Germany, 1995. Springer-Verlag.

[Fisher and Owens, 1995b] M. Fisher and R. Owens, editors. *Executable Modal and Temporal Logics*, volume 897 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, February 1995.

[Fisher and Wooldridge, 1997] M. Fisher and M. Wooldridge. On the Formal Specification and Verification of Multi-Agent Systems. *International Journal of Cooperative Information Systems*, 6(1):37–65, January 1997.

[Fisher *et al.*, 2001] M. Fisher, C. Dixon, and M. Peim. Clausal Temporal Resolution. *ACM Transactions on Computational Logic*, 2(1):12–56, January 2001.

[Fisher *et al.*, 2003] M. Fisher, C. Ghidini, and B. Hirsch. Organising logic-based agents. In M. Hinchey, J. Rash, W. Truszkowski, C. Rouff, and D. Gordon-Spears, editors, *Proceedings of Second International Workshop on Formal Approaches to Agent-Based Systems (FAABS)*, volume 2699 of *Lecture Notes in Computer Science*, pages 15–27, Greenbelt, USA, October 2003. Springer.

[Fisher, 1991] M. Fisher. A Resolution Method for Temporal Logic. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI)*, Sydney, Australia, 1991. Morgan Kaufman.

[Fisher, 1993] M. Fisher. Concurrent METATEM — A Language for Modeling Reactive Systems. In *Parallel Architectures and Languages, Europe (PARLE)*, volume 694 of *Lecture Notes in Computer Science*, Munich, Germany, June 1993. Springer-Verlag.

[Fisher, 1994] M. Fisher. A Survey of Concurrent METATEM — The Language and its Applications. In *First International Conference on Temporal Logic (ICTL)*, volume 827 of *Lecture Notes in Computer Science*, Bonn, Germany, July 1994. Springer-Verlag.

[Fisher, 1996a] M. Fisher. A Temporal Semantics for Concurrent METATEM. *Journal of Symbolic Computation*, 22(5/6), November/December 1996.

[Fisher, 1996b] M. Fisher. An Introduction to Executable Temporal Logics. *Knowledge Engineering Review*, 11(1):43–56, March 1996.

[Fisher, 1997a] M. Fisher. A Normal Form for Temporal Logic and its Application in Theorem-Proving and Execution. *Journal of Logic and Computation*, 7(4), July 1997.

[Fisher, 1997b] M. Fisher. Implementing BDI-like Systems by Direct Execution. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. Morgan-Kaufmann, 1997.

[Fisher, 2004] M. Fisher. Temporal Development Methods for Agent-Based Systems. To appear in *Journal of Autonomous Agents and Multi-Agent Systems*, 2004.

[Fitting, 1983] M. Fitting. *Proof Methods for Modal and Intuitionistic Logics*. Reidel, 1983.

[Forbus, 1989] K. Forbus. Introducing Actions into Qualitative Simulation. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1273–1278. Morgan Kaufmann, 1989.

[Fox and Long, 1996] M. Fox and D. Long. An Efficient Algorithm for Managing Partial Orders in Planning. *ACM SIGART Bulletin*, 7(4):3–12, 1996.

[Fox and Long, 2002a] M. Fox and D. Long. PDDL+ : Planning with Time and Metric Resources. Technical Report Department of Computer Science, Durham University, UK, 2002.

[Fox and Long, 2002b] M. Fox and D. Long. PDDL+: Modelling Continuous Time-dependent Effects. In *Proceedings of the Third International NASA Workshop on Planning and Scheduling for Space*, 2002.

[Fox and Long, 2003] M. Fox and D. Long. PDDL2.1: An extension to PDDL for expressing temporal planning domain s. *Journal of AI Research*, 20, 2003.

[Fox *et al.*, 1998] J. Fox, N. Johns, and A. Rahmanzadeh. Disseminating medical knowledge: the pro*forma* approach. *Artificial Intelligence in Medicine*, 14:157–181, 1998.

[Franceschet and Montanari, 2001a] M. Franceschet and A. Montanari. A Combined Approach to Temporal Logics for Time Granularity. In *Proceedings of the Second International Workshop on Methods for Modalities (M4M)*, 2001.

[Franceschet and Montanari, 2001b] M. Franceschet and A. Montanari. Towards an Automata-Theoretic Counterpart of Combined Temporal Logics. In *Proceedings of the Second International Workshop on Verification and Computational Logic*, pages 55–74, 2001.

[Franceschet and Montanari, 2003] M. Franceschet and A. Montanari. Branching Within Time: An Expressively Complete and Elementarily Decidable Temporal Logic for Time Granularity. *Research on Language and Computation*, 1(3-4):229–263, 2003.

[Franceschet and Montanari, 2004] M. Franceschet and A. Montanari. Temporalized Logics and Automata for Time Granularity. *Theory and Practice of Logic Programming*, 4(5-6):621–658, 2004.

[Franceschet *et al.*, 2003] M. Franceschet, A. Montanari, A. Peron, and G. Sciavicco. Definability and Decidability of Binary Predicates for Time Granularity. In *Proceedings of the Tenth International Symposium on Temporal Representation and Reasoning and of the Fourth International Conference on Temporal Logic (TIME-ICTL)*, pages 192–202. IEEE Computer Society Press, 2003.

[Franceschet *et al.*, 2004] M. Franceschet, A. Montanari, and M. de Rijke. Model Checking for Combined Logics with an Application to Mobile Systems. *Automated Software Engineering*, 11(3):287–319, 2004.

[Franceschet, 2002] M. Franceschet. *Dividing and Conquering the Layered Land*. PhD thesis, Department of Mathematics and Computer Science, University of Udine, Udine, Italy, 2002.

[Franconi and Toman, 2003] E. Franconi and D. Toman. Fixpoint Extensions of Temporal Description Logics. In *Proceedings of Workshop on Description Logics (DL)*, volume 81 of *CEUR-WS*, pages 160–167, 2003.

[Franklin and Graesser, 1996] S. Franklin and A. Graesser. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents . In J. P. Müller, M. J. Wooldridge, and N. R. Jennings, editors, *Intelligent Agents III — Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages (ATAL)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Heidelberg, 1996.

[Frege, 1972] G. Frege. *Conceptual Notation (Begriffschrift), and Related Articles*. Oxford : Clarendon Press, 1972. (English translation, edited by T. Bynum.).

[Freksa, 1992] Christian Freksa. Temporal reasoning based on semi-intervals. *Artificial intelligence*, 54(1):199–227, 1992.

[Friedman *et al.*, 1998] N. Friedman, K. Murphy, and S. Russell. Learning the Structure of Dynamic Probabilistic Networks. In *Proceedings of International Conference on Uncertainty in AI (UAI)*, pages 139–147, 1998.

[Fries, 1972] J.F. Fries. Time oriented patient records and a computer databank. *Journal of the American Medical Association*, 222:1536–1543, 1972.

[Früehwirth, 1996] T. Früehwirth. Temporal Annotated Constraint Logic Programming. *Journal of Symbolic Computation*, 22(5/6), 1996.

[Fung and Kowalski, 1997] T.H. Fung and R. Kowalski. The IFF Proof Procedure for Abductive Logic Programming. *Journal of Logic Programming*, 33(2):151–165, 1997.

[Furer, 1982] M. Furer. The Complexity of Presburger Arithmetic with Bounded Quantifier Alternation Depth. *Theoretical Computer Science*, 18:105–111, 1982.

[Fusaoka, 1996] A. Fusaoka. Situation Calculus on a Dense Flow of Time. In *Proceedings of AAAI Conference*, pages 633–638. AAAI press, 1996.

[Gabbay and Hodkinson, 1990] D. M. Gabbay and I. M. Hodkinson. An Axiomatisation of the Temporal Logic with Until and Since over the Real Numbers. *Journal of Logic and Computation*, 1(2):229 – 260, 1990.

[Gabbay *et al.*, 1980] D. Gabbay, A. Pnueli, S. Shelah, and J. Stavi. The Temporal Analysis of Fairness. In *Proceedings of the Seventh ACM Symposium on the Principles of Programming Languages (POPL)*, pages 163–173, Las Vegas, Nevada, January 1980.

[Gabbay *et al.*, 1994a] D. Gabbay, I. Hodkinson, and M. Reynolds. *Temporal Logic: Mathematical Foundations and Computational Aspects, Volume 1*. Oxford University Press, 1994.

[Gabbay *et al.*, 1994b] D.M. Gabbay, C. Hogger, J. Robinson, and D. Nute. *Handbook of Logic in AI and Logic Programming - Volume 4: Epistemic and Temporal Reasoning*. Clarendon Press, Oxford, 1994.

[Gabbay *et al.*, 2000] D.M. Gabbay, M.A. Reynolds, and M. Finger. *Temporal Logic: Mathematical Foundations and Computational Aspects - Volume 2*, volume 40 of *Oxford Logic Guides*. Oxford University Press, 2000.

[Gabbay, 1981] D.M. Gabbay. The Separation Theorem for Temporal Logic. Technical Report DFG Project RO/245/12, University of Stuttgart, 1981.

[Gal *et al.*, 1994] A. Gal, O. Etzioni, and A. Segev. Representation of Highly-Complex Knowledge in a Database. *Journal of Intelligent Information Systems*, 3(2):185–203, 1994.

[Galipienso and Sanchis, 2002] M.I.A. Galipienso and F.B. Sanchis. Representation and Reasoning with Disjunction Temporal Constraints. In *Proceedings of Ninth International Symposium on Temporal Representation and Reasoning (TIME)*, 2002.

[Galton, 1984] A. P. Galton. *The Logic of Aspect: an Axiomatic Approach*. Clarendon Press, Oxford, 1984.

[Galton, 1987] A. Galton, editor. *Temporal Logics and Their Applications*. Academic Press, 1987.

[Galton, 1990] Antony Galton. A Critical Examination of Allen's Theory of Action and Time. *Artificial Intelligence*, 42:159–188, 1990.

[Galton, 1991] A. P. Galton. Reified Temporal Theories and How To Unreify Them. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1177–1182. Morgan Kaufmann, 1991.

[Galton, 1996a] A. P. Galton. A Note on a Lemma of Ladkin. *Journal of Logic and Computation*, 6(1):1–4, 1996.

[Galton, 1996b] A. P. Galton. An Investigation of 'Non-Intermingling' Principles in Temporal Logic. *Journal of Logic and Computation*, 6, 2:271–294, 1996.

[Gamper and Nejdl, 1997] J. Gamper and W. Nejdl. Abstract Temporal Diagnosis in Medical Domains. *Artificial Intelligence in Medicine*, 10:209–234, 1997.

[Garagnani, 2000] M. Garagnani. A Correct Algorithm for Efficient Planning with Preprocessed Domain Axioms. In M. Bramer, A. Preece, and F. Coenen, editors, *Research and Development in Intelligent Systems XVII (Proceedings of ES'2000)*. Springer-Verlag, 2000.

[Gardner, 1987] A. Gardner. *An Artificial Intelligence Approach to Legal Reasoning*. MIT Press, 1987.

[Garrido *et al.*, 2002] A. Garrido, M. Fox, and D. Long. Temporal Planning with PDDL2.1. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 2002.

[Gayral, 1992] F. Gayral. *Sémantique du Langage Naturel et Profondeur Variable: Une Première Approche*. PhD thesis, Université de Paris-Nord, Villetaneuse (FR), 1992.

[Gazen and Knoblock, 1997] B. Gazen and C. Knoblock. Combining the Expressivity of UCPOP with the Efficiency of Graphplan. In *Proceedings of European Conference on Planning (ECP)*, pages 221–233, 1997.

[Geerts *et al.*, 2001] F. Geerts, S. Haesevoets, and B. Kuijpers. A Theory of Spatio-Temporal Database Queries. In *Proceedings of the International Workshop on Database Programming Languages*, pages 198–212, 2001.

[Gelfond and Lifschitz, 1988] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. Kowalski and K. Bowen, editors, *Logic Programming: Proc. of the Fifth Int'l Conf. and Symp.*, pages 1070–1080. MIT Press, 1988.

[Gelfond and Lifschitz, 1990] M. Gelfond and V. Lifschitz. Logic Programs with Classical Negation. In D. Warren and P. Szeredi, editors, *Proceedings of the Seventh International Conference on Logic Programming (ICLP)*, pages 579–597, 1990.

[Gelfond and Lifschitz, 1991] M. Gelfond and V. Lifschitz. Classical Negation in Logic Programs and Disjunctive Databases. *New Generation Computing*, 9:365–387, 1991.

[Gelfond and Lifschitz, 1993] M. Gelfond and V. Lifschitz. Representing Actions and Change by Logic Programs. *Journal of Logic Programming*, 17(2,3,4):301–323, 1993.

[Gelfond *et al.*, 1991] M. Gelfond, V. Lifschitz, and A. Rabinov. What are the Limitations of the Situation Calculus? In R. Boyer, editor, *Essays in honor of Woody Bledsoe*, pages 167–179. Kluwer Academic, 1991.

[Gelfond *et al.*, 2001] M. Gelfond, M. Balduccini, and J. Galloway. Diagnosing Physical Systems in A-Prolog. In T. Eiter, W. Faber, and M. Truszczyński, editors, *Proceedings of the Sixth International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, pages 213–226, 2001.

[Gelfond, 1994] M. Gelfond. Logic Programming and Reasoning with Incomplete Information. *Annals of Mathematics and Artificial Intelligence*, 12:19–116, 1994.

[Gelfond, 2001] M. Gelfond. Representing knowledge in A-Prolog. In *Computational Logic: from Logic Programming to the Future, Collection of Papers in Honour of B. Kowalski*. Elsevier, 2001.

[Gentzen, 1934] G. Gentzen. Untersuchungen über das logische schliessen. *Math. Zeitschrift*, 39:176–210, 405–431, 1934.

[Gerevini and Cristani, 1995] A. Gerevini and M. Cristani. Reasoning with Inequations in Temporal Constraint Networks. Technical report, IRST - Instituto per la Ricerca Scientifica e Tecnologica, Povo TN, Italy, 1995. (A shorter version appears in the Proceedings of the Workshop on Spatial and Temporal Reasoning, IJCAI-95).

[Gerevini and Cristani, 1996] A. Gerevini and M. Cristani. On Temporal Constraint Networks with Inequations. Technical Report RT 199611-3, Dipartimento di Elettronica per l'Automazione, Università di Brescia, Italy, 1996.

[Gerevini and Cristani, 1997] A. Gerevini and M. Cristani. On Finding Solutions in Temporal Constraint Networks. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1460–1465, Nagoya, Japan, 1997. Morgan Kaufmann.

[Gerevini and Nebel, 2002] A. Gerevini and B. Nebel. Qualitative Spatio-Temporal Reasoning with RCC-8 and Allen's Interval Calculus: Computational Complexity. In *Proceedings of the Seventh European Conference on Artificial Intelligence (ECAI)*, pages 312–316. IOS Press, 2002.

[Gerevini and Renz, 1998] A. Gerevini and J. Renz. Combining Topological and Qualitative Size Constraints for Spatial Reasoning. In *Proceedings of the Fourth International Conference on Principles and Practice of Constraint Programming (CP)*. Splinger Verlag, 1998.

[Gerevini and Schubert, 1994a] A. Gerevini and L. Schubert. An Efficient Method for Managing Disjunctions in Qualitative Temporal Reasoning. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 215–225, San Francisco, CA, 1994. Morgan Kaufmann.

[Gerevini and Schubert, 1994b] A. Gerevini and L. Schubert. On Point-Based Temporal Disjointness. *Artificial Intelligence*, 70:347–361, 1994.

[Gerevini and Schubert, 1995a] A. Gerevini and L. Schubert. Efficient Algorithms for Qualitative Reasoning about Time. *Artificial Intelligence*, 74(2):207–248, 1995.

[Gerevini and Schubert, 1995b] A. Gerevini and L. Schubert. On Computing the Minimal Labels in Time Point Algebra Networks. *Computational Intelligence*, 11(3):443–448, 1995.

[Gerevini and Serina, 2002] A. Gerevini and I. Serina. LPG: A Planner Based on Local Search for Planning Graphs. In *Proceedings of the Sixth International Conference on AI Planning Systems (AIPS)*. AAAI Press, 2002.

[Gerevini *et al.*, 1993] A. Gerevini, L. Schubert, and S. Schaeffer. Temporal Reasoning in Timegraph I-II. *SIGART Bulletin*, 4(3):21–25, 1993.

[Gerevini *et al.*, 1995] A. Gerevini, L. Schubert, and S. Schaeffer. The Temporal Reasoning Tools TimeGraph-I-II. *International Journal of Artificial Intelligence Tools*, 4(1–2):281–299, 1995. (A shorter version appeared in *Proceedings of the Sixth IEEE Int. Conf. on Tools with Artificial Intelligence*, p. 513–520, IEEE Computer Society Press, 1994; and in *SIGART Bulletin*, ACM Press, 4(3):21-25, July 1993.).

[Gerevini *et al.*, 1996] A. Gerevini, A. Perini, and F. Ricci. Incremental Algorithms for Managing Temporal Constraints. In *Proceedings of the Eighth IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, Toulouse, France, 1996. IEEE Computer Society Press.

[Gerevini, 1997] A. Gerevini. Reasoning About Time and Actions in Artificial Intelligence: Major Issues. In O. Stock, editor, *Spatial and Temporal Reasoning*, pages 43–70. Kluwer Academic Publishers, 1997.

[Gerevini, 2003a] A. Gerevini. Incremental Tractable Reasoning about Qualitative Temporal Constraints. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1283–1288. Morgan Kaufmann, 2003.

[Gerevini, 2003b] A. Gerevini. Incremental Tractable Reasoning about Qualitative Temporal Information. Technical Report 2003-12-32, Dipartimento di Elettronica per l'Automazione, Università di Brescia, Italy, 2003.

[Ghahramani and Jordan, 1996] Z. Ghahramani and M.I. Jordan. Factorial hidden Markov models. In *Proceedings of NIPS*, pages 396–402, 1996.

[Ghallab and Laruelle, 1994] M. Ghallab and H. Laruelle. Representation and Control in IxTeT, a Temporal Planner. In *Proceedings of the International Conference on Artificial Intelligence Planning and Scheduling (AIPS)*, 1994.

[Ghallab and Mounir Alaoui, 1989] M. Ghallab and A. Mounir Alaoui. Managing Efficiently Temporal Relations Through Indexed Spanning Trees. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, pages 1297–1303, San Mateo, CA, 1989. Morgan Kaufmann.

[Ghallab, 1996] M. Ghallab. On Chronicles: Representation, On-line Recognition and Learning. In *Proceedings of Interntation Conference on Knowledge Representation and Reasoning (KR)*, pages 597–606, 1996.

[Giannotti *et al.*, 2003] F. Giannotti, G. Manco, and J. Wijsen. Logical Languages for Data Mining. In Chomicki et al. [2003b], chapter 9, pages 325–361.

[Gilks and Berzuini, 2001] W.R. Gilks and C. Berzuini. Following a Moving Target — Monte Carlo Inference for Dynamic Bayesian Systems. *Journal of the Royal Statistical Society (Series B)*, 63(1):127–146, 2001.

[Ginsberg and Smith, 1988a] M. Ginsberg and D. Smith. Reasoning about Action I: A Possible Worlds Approach. *Artificial Intelligence*, 35(2):165–196, June 1988.

[Ginsberg and Smith, 1988b] M. Ginsberg and D. Smith. Reasoning about Action II: The Qualification Problem. *Artificial Intelligence*, 35(3):311–342, July 1988.

[Ginsburg and Hull, 1983] S. Ginsburg and R. Hull. Order Dependency in the Relational Model. *Theoretical Computer Science*, 26:149–195, 1983.

[Ginsburg and Hull, 1986] S. Ginsburg and R. Hull. Sort Sets in the Relational Model. *Journal of the ACM*, 33(3):465–488, 1986.

[Giunchiglia and Lifschitz, 1998] E. Giunchiglia and V. Lifschitz. An Action Language Based on Causal Explanation: Preliminary Report. In *Proceeings of the AAAI Conference*, pages 623–630, 1998.

[Giunchiglia *et al.*, 1997] F. Giunchiglia, A. Villafiorita, and T. Walsh. Theories of Abstraction. *AI Communications*, 10(2):167–176, 1997.

[Goldblatt, 1987] R. Goldblatt. *Logics of Time and Computation*, volume 7 of *CSLI Lecture Notes*. Center for the Study of Language and Information, Stanford University, California, 1987.

[Goldblatt, 1991] R. Goldblatt. The McKinsey Axiom is not Canonical. *The Journal of Symbolic Logic*, 56(2):554–562, 1991.

[Goldin, 1997] D.Q. Goldin. *Constraint Query Algebras*. PhD thesis, Dept. of Computer Science, Brown University, USA, 1997.

[Golumbic and Shamir, 1993] Martin Charles Golumbic and Ron Shamir. Complexity and algorithms for reasoning about time: A graph-theoretic approach. *Journal of the ACM*, 40(5):1108–1133, 1993.

[Goodwin *et al.*, 1994] S.D. Goodwin, E. Neufeld, and A. Trudel. Probabilistic Temporal Representation and Reasoning. *International Journal of Expert Systems*, 7(3):261–289, 1994.

[Gordon and Veloso, 1996] C. Gordon and M. Veloso. The PRESTIGE Project: Implementing Guidelines in Healthcare. In *Medical Informatics Europe*, pages 887–891. IOS Press, 1996.

[Goré, 1997] R. Goré. Tableau Methods for Modal and Temporal Logics. Technical Report TR-ARP-15-95 (revised 1997), Automated Reasoning Project, Australian National University, Canberra, Australia, 1997.

[Gough, 1984] G. D. Gough. Decision Procedures for Temporal Logic. Master's thesis, Department of Computer Science, University of Manchester, October 1984. (Also University of Manchester, Department of Computer Science, Technical Report UMCS-89-10-1.).

[Grahne, 1991] G. Grahne. *The Problem of Incomplete Information in Relational Databases*, volume 554 of *Lecture Notes in Computer Science*. Springer Verlag, 1991.

[Gregersen and Jensen, 1999] H. Gregersen and C. S. Jensen. Temporal Entity-Relationship Models - A Survey. *Knowledge and Data Engineering*, 11(3):464–497, 1999.

[Grimshaw and Russel, 1993] J.M. Grimshaw and I.T. Russel. Effects of Clinical Guidelines on Medical Practice: A Systematic Review of Rigorous Evaluation. *Lancet*, pages 1317–1322, 1993.

[Gruska, 1990] J. Gruska. Synthesis, Structure and Power of Systolic Computations. *Theoretical Computer Science*, 71(1):47–77, 1990.

[Guesgen, 1989] H.-W. Guesgen. Spatial Reasoning Based on Allen's Temporal Logic. Technical Report TR-89-094, ICSI, 1989.

[Gupta *et al.*, 1997] V. Gupta, T.A. Henziner, and R. Jagadeesan. Robust timed automata. In *Hybrid and Real-time Systems (HART)*, volume 1201 of *Lecture Notes in Computer Science*, pages 331–345. Springer-Verlag, 1997.

[Gurevich and Shelah, 1985] Y. Gurevich and S. Shelah. The Decision Problem for Branching Time Logic. *J. of Symbolic Logic*, 50:668–681, 1985.

[Haarslev *et al.*, 1998] V. Haarslev, C. Lutz, and R. Möller. Foundation of Spatiotemporal Reasoning with Description Logics. In A. Cohn, L. Schubert, and S. C. Shapiro, editors, *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 112–123, Trento, Italy, June 1998. Morgan Kaufmann.

[Haddawy, 1994] P. Haddawy. *Representing Plans under Uncertainty: A Logic of Time, Chance, and Action*. Number 770 in Lecture Notes in Artificial Intelligence. Springer-Verlag, 1994.

[Hafer and Thomas, 1987] T. Hafer and W. Thomas. Computation tree logic CTL* and path quantifiers in the monadic theory of the binary tree. In *Proceedings of the International Colloquium on Automata, Languages and Programming (ICALP)*, volume 267 of *Lecture Notes in Computer Science*, pages 269–279, Karlsruhe, Germany, 1987. Springer.

[Haimowitz and Kohane, 1996] I.J. Haimowitz and I.S. Kohane. Managing Temporal Worlds for Medical Trend Diagnosis. *Artificial Intelligence in Medicine*, 8(3):299–321, 1996.

[Halpern and Moses, 1985] J. Y. Halpern and Y. Moses. A Guide to the Modal Logics of Knowledge and Belief. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 480–490, 1985.

[Halpern and Moses, 1992] J. Y. Halpern and Y. Moses. A Guide to Completeness and Complexity for Modal Logics of Knowledge and Belief. *Artificial Intelligence*, 54:319–379, 1992.

[Halpern and Shoham, 1986] J. Halpern and Y. Shoham. A Propositional Modal Logic of Time Intervals. In *Proceedings of International Symposium on Logic in Computer Science (LICS)*. IEEE Press, 1986.

[Halpern and Shoham, 1991] J. Y. Halpern and Y. Shoham. A Propositional Modal Logic of Time Intervals. *Journal of the ACM*, 38(4):935–962, 1991.

[Halpern and Vardi, 1991] J. Y. Halpern and M. Y. Vardi. Model Checking vs. Theorem Proving: A Manifesto. In V. Lifschitz, editor, *AI and Mathematical Theory of Computation — Papers in Honor of John McCarthy*. Academic Press, 1991.

[Hamblin, 1971] C. L. Hamblin. Instants and Intervals. *Studium Generale*, 24:127–34, 1971.

[Hamblin, 1972] C. L. Hamblin. Instants and Intervals. In J. T. Fraser, F. C. Haber, and G. H. Müller, editors, *The Study of Time*, pages 324–328. Springer-Verlag, New York, 1972.

[Hanks and McDermott, 1986] S. Hanks and D. McDermott. Default Reasoning, Nonmonotonic Logic and the Frame Problem. In *Proceedings of the American Association for AI Conference (AAAI)*, pages 328–333, 1986.

[Hanks and McDermott, 1987] S. Hanks and D. McDermott. Nonmonotonic Logic and Temporal Projection. *Artificial Intelligence*, 33(3):379–412, 1987.

[Hanks and McDermott, 1994] S. Hanks and D. McDermott. Modeling a Dynamic and Uncertain World I: Symbolic and Probabilistic Reasoning about Change. *Artificial Intelligence*, 65(2), 1994.

[Hanks *et al.*, 1995] S. Hanks, D. Madigan, and J. Gavrin. Probabilistic Temporal Reasoning with Endogenous Change. In *Proceedings of International Conference on Uncertainty in Artificial Intelligence (UAI)*, 1995.

[Hanks, 1990] S. Hanks. Projecting Plans for Uncertain Worlds. Technical Report 756, Department of Computer Science, Yale University, USA, January 1990. Ph.D. thesis.

[Hansen and Zilberstein, 1996] E.A. Hansen and S. Zilberstein. Monitoring the Progress of Anytime Problem-Solving. In *Proceedings of the Thirteenth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 1229–1234. AAAI Press/The MIT Press, 1996.

[Harel *et al.*, 1980] D. Harel, D. Kozen, and R. Parikh. Process Logic: Expressiveness, Decidability, Completeness. In *Proceedings of IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 129–142, 1980.

[Harel *et al.*, 1982] D. Harel, D. Kozen, and R. Parikh. Process Logic: Expressiveness, Decidability, Completeness. *Journal of Computing System Sciences*, 25:144–170, 1982.

[Harel, 1979] D. Harel. *First-Order Dynamic Logic*, volume 68. Springer-Verlag Inc., New York, USA, 1979.

[Harel, 1984] D. Harel. Dynamic Logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume II: Extensions of Classical Logic, pages 497–604. D. Reidel Publishing Co., Dordrecht, 1984.

[Haslum and Geffner, 2001] P. Haslum and H. Geffner. Heuristic Planning with Time and Resources. In *Proceedings of the European Conference on Planning (ECP)*, Toledo, Spain, 2001.

[Haugh, 1987] B. A. Haugh. Non-Standard Semantics for The Method of Temporal Arguments. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, pages 449–454. Morgan Kaufmann, 1987.

[Heckerman and Miller, 1986] D.E. Heckerman and R.A. Miller. Towards a Better Understanding of the INTERNIST-1 Knowledge Base. In R. Salamon, B. Blum, and M. Jorgensen, editors, *Proceedings of the Fifth Conference on Medical Informatics (MEDINFO)*, pages 27–31, New York, 1986. North-Holland.

[Hendry and Richard, 1990] D.F. Hendry and J.F. Richard. Likelihood Evaluation for Dynamic Latent Variables Models. In H.M. Amann, D.A. Belsley, and L.F. Pau, editors, *Computational Economics and Econometrics*, chapter 1. Kluwer, Dordrecht, 1990.

[Henkin, 1949] L. Henkin. The Completeness of the First-Order Functional Calculus. *J. of Symbolic Logic*, 14:159–166, 1949.

[Henzinger *et al.*, 1994] T. Henzinger, Z. Manna, and A. Pnueli. Temporal Proof Methodologies for Timed Transition Systems. *Information and Computation*, 112:273–337, 1994.

[Henzinger *et al.*, 1995] T.A. Henzinger, P-H. Ho, and H Wong-Toi. A user guide to HYTECH. In E. Brinksma, W.R. Cleaveland, K.G. Larsen, T. Margaria, and B. Steffen, editors, *Proceedings of Conference on Tools and Algorithms for the Construction and Analysis of Systems: (TACAS)*, volume 1019 of *Lecture Notes in Computer Science*, pages 41–71, 1995.

[Henzinger, 1996] T.A. Henzinger. The Theory of Hybrid Automata. In *Proceedings of the 11th Annual Symposium on Logic in Computer Science (LICS)*, pages 278–292. IEEE Computer Society Press, 1996.

[Herbert *et al.*, 1995] S.I. Herbert, C.J. Gordon, A. Jackson–Smale, and J.L. Renaud Salis. Protocols for Clinical Care. *Computer Methods and Programs in Biomedicine*, 48:21–26, 1995.

[Heuerding *et al.*, 1995] A. Heuerding, G. Jaeger, S. Schwendimann, and M. Seyfried. Propositional Logics on the Computer. In P. Baumgartner, R. Hähnle, and J. Posegga, editors, *Proceedings of Tableaux Workshop*, volume 918 of *Lecture Notes in Computer Science*, pages 310–323. Springer-Verlag, 1995.

[Hintikka, 1955] J. Hintikka. Form and Content in Quantification Theory. *Acta Philosophica Fennica*, 8:7–55, 1955.

[Hirsh, 1996] R. Hirsh. Relation Algebras of Intervals. *Artificial Intelligence*, 83(2):267–295, 1996.

[Hobbs, 1985] J. Hobbs. Granularity. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 432–435, 1985.

[Hochbaum and Naor, 1994] D.S. Hochbaum and J. Naor. Simple and Fast Algorithms for Linear and Integer Programs with Two Variables per Inequality. *SIAM Journal on Computing*, 23(6):1179–1192, 1994.

[Hodges, 1984] W. Hodges. Elementary Predicate Logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*. D. Reidel, Dordrecht, 1984.

[Hodkinson *et al.*, 2000] I. Hodkinson, F. Wolter, and M. Zakharyashev. Decidable Fragments of First-Order Temporal Logics. *Annals of Pure and Applied Logic*, 106:85–134, 2000.

[Hodkinson *et al.*, 2002] I. Hodkinson, F. Wolter, and M. Zakharyaschev. Decidable and Undecidable Fragments of First-Order Branching Temporal Logics. In *IEEE Symposium on Logic in Computer Science*, pages 393–402, 2002.

[Hodkinson *et al.*, 2003] I. Hodkinson, R. Kontchakov, A. Kurucz, F. Wolter, and M. Zakharyaschev. On the Computational Complexity of Decidable Fragments of First-order Linear Temporal Logics. In *Proceedings of the Joint Tenth International Symposium on Temporal Representation and Reasoning and Fourth International Conference on Temporal Logic (TIME-ICTL)*, 2003.

[Hodkinson, 2000] I. Hodkinson. Temporal Logic and Automata (Chapter 2). In *Temporal Logic: Mathematical Foundations and Computational Aspects, Vol. 2*. Oxford University Press, 2000.

[Hodkinson, 2002] I. M. Hodkinson. Monodic Packed Fragment with Equality is Decidable. *Studia Logica*, 72(2):185–197, 2002.

[Hoffmann and Nebel, 2000] J. Hoffmann and B. Nebel. The FF Planning System: Fast Plan Generation Through Heuristic Search. *Journal of AI Research*, 14:253–302, 2000.

[Hogge, 1987] J.C. Hogge. TPLAN: A Temporal Interval-based Planner with Novel Extensions. Technical Report UIUCDCS-R-87, University of Illinois, USA, 1987.

[Holldobler and Thielscher, 1993] S Holldobler and M Thielscher. Actions and Specificity. In D. Miller, editor, *Proceedings of the International Conference on Logic Programming (ICLP)*, pages 164–180, 1993.

[Hollunder *et al.*, 1990] B. Hollunder, W. Nutt, and M. Schmidt-Schaus. Subsumption Algorithms for Concept Description Languages. In *Proceedings of European Conference on Artificial Intelligence (ECAI)*, pages 348–353, 1990.

[Holzmann, 1997] G.J. Holzmann. The Model Checker Spin. *IEEE Transactions on Software Engineering*, 23(5):279–295, May 1997.

[Holzmann, 2003] G. J. Holzmann. *The Spin Model Checker: Primer and Reference Manual*. Addison-Wesley, November 2003.

[Horn *et al.*, 1997] W. Horn, S. Miksch, G. Egghart, C. Popow, and F. Paky. Effective Data Validation of High–Frequency Data: Time–Point–, Time–Interval–, and Trend–Based Methods. *Computers in Biology and Medicine*, 27(5):389–409, 1997.

[Horn, 2001] W. Horn. AI in Medicine on its way from Knowledge-Intensive to Data-Intensive Systems. *Artificial Intelligence in Medicine*, 23:3–12, 2001.

[Horrocks, 1998] I. Horrocks. The FaCT System. In *Proceedings of International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (Tableaux'98)*, volume 1397 of *Lecture Notes in Computer Science*, pages 307–312. Springer-Verlag, May 1998.

[Howey and Long, 2002] R. Howey and D. Long. Validating Plans with Continuous Effects. Technical report, Department of Computer Science, University of Durham, UK, 2002.

[Hripcsak *et al.*, 1994] G. Hripcsak, P. Ludemann, T.A. Pryor, O.B. Wigertz, and P.D. Clayton. Rationale for the Arden Syntax. *Computers and Biomedical Research*, 27:291–324, 1994.

[Hrycej, 1993] T. Hrycej. A Temporal Extension of Prolog. *Journal of Logic Programming*, 15:113–145, 1993.

[Hughes and Cresswell, 1968] G. Hughes and M. Cresswell. *An Introduction to Modal Logic*. Methuen, London, 1968.

[Hustadt and Konev, 2002] U. Hustadt and B. Konev. TRP++: A Temporal Resolution Prover. In *Proceedings of 3rd International Workshop on the Implementation of Logics*, Tbilisi, Georgia, October 2002.

[Hustadt and Konev, 2003] U. Hustadt and B. Konev. TRP 2.0: A Temporal Resolution Prover. In *Proceedings of 19th International Conference on Automated Deduction (CADE)*, LNCS. Springer, July/August 2003.

[Hustadt and Schmidt, 2002] U. Hustadt and R. A. Schmidt. Scientific Benchmarking with Temporal Logic Decision Procedures. In D. Fensel, F. Giunchiglia, D. McGuinness, and M.-A. Williams, editors, *Proceedings of the Eighth International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 533–544. Morgan Kaufmann, 2002.

[Hustadt *et al.*, 2000] U. Hustadt, C. Dixon, R. A. Schmidt, and M. Fisher. Normal Forms and Proofs in Combined Modal and Temporal Logics. In *Proceedings of the Third International Workshop on Frontiers of Combining Systems (FroCoS)*, volume 1794 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, March 2000.

[Hwang and Schubert, 1994] C.L. Hwang and L.K. Schubert. Interpreting Tense, Aspect, and Time Adverbials: A Compositional, Unified Approach. In D.M. Gabbay and H.J. Ohlbach, editors, *Proceedings of the First International Conference on Temporal Logic (ICTL)*, volume 827 of *Lecture Notes in Computer Science*, pages 237–264, Berlin, 1994. Springer-Verlag.

[Imielinski and Lipski, 1984] T. Imielinski and W. Lipski. Incomplete Information in Relational Databases. *Journal of ACM*, 31(4):761–791, 1984.

[Immerman and Kozen, 1989] N. Immerman and D. Kozen. Definability with Bounded Number of Bound Variables. *Information and Computation*, 83(2):121–139, 1989.

[ISO, 1992] ISO. Database Language SQL. ISO/IEC 9075:1992, International Organization for Standardization, 1992.

[Iwasaki and Low, 1992] Y. Iwasaki and C.M. Low. Device Modelling Environment: An Integrated Model-formulation and simulation environment for Continuous and Discrete Phenomena. In *Proceedings of the Conference on Intelligent Systems Engineering*, 1992.

[Iwasaki *et al.*, 1995] Y. Iwasaki, A. Farquhar, V. Saraswat, D. Bobrow, and V. Gupta. Modelling Time in Hybrid Systems: How Fast Is 'Instantaneous'? In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1773–1781. Morgan Kaufmann, 1995.

[Jackendoff, 1976] R. Jackendoff. Toward an Explanatory Semantic Representation. *Linguistic Inquiry*, 7(1):89–150, 1976.

[Jaffar and Lassez, 1987] J. Jaffar and J-L. Lassez. Constraint Logic Programming. In *Proceeidngs of ACM Symposium on Principles of Programming Languages (POPL)*, 1987.

[Jaffar and Maher, 1994] J. Jaffar and M.J. Maher. Constraint Logic Programming: A Survey. *Journal of Logic Programming*, 19(20):503–581, 1994.

[Jaffar *et al.*, 1994] J. Jaffar, M. J. Maher, P. Stuckey, and R. Yap. Beyond Finite Domains. In A. Borning, editor, *Proceedings of PPCP Conference*, volume 874 of *Lecture Notes in Computer Science*, pages 86–94. Springer Verlag, 1994.

[Jäger *et al.*, 2002] G. Jäger, P. Balsiger, A. Heuerding, S. Schwendimann, M. Bianchi, K. Guggisberg, G. Janssen, W. Heinle, F. Achermann, A. D. Boroumand, P. Brambilla, I. Bucher, and H. Zimmermann. LWB–The Logics Workbench 1.1. `http://www.lwb.unibe.ch/`, 2002. University of Berne, Switzerland.

[Janssen, 1999] G.L.J.M. Janssen. *Logics for Digital Circuit Verification: Theory, Algorithms, and Applications*. PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, 1999.

[Jennings and Wooldridge, 1998] N. R. Jennings and M. Wooldridge. Applications of Agent Technology. In *Agent Technology: Foundations, Applications, and Markets*. Springer-Verlag, Heidelberg, 1998.

[Jensen *et al.*, 1993] C.S. Jensen, M.D. Soo, and R.T. Snodgrass. Unification of Temporal Data Models. In *Proceedings of IEEE International Conference on Data Engineering*, 1993.

[Jensen *et al.*, 1994] C. Jensen, J. Clifford, S. Elmasri, R. Gadia, P. Hayes, S. Jajodia, C. Dyreson, F. Grandi, W. Kaefer, N. Kline, N. Lorentzos, Y. Mitsopoulos, A. Montanari, D. Nonen, E. Peressi, B. Pernici, J. Roddick, N. Sarda, M. Scalas, A. Segev, Richard Snodgrass, Mike Soo, A. Tansel, P. Tiberio, and W. Gio. A Consensus Glossary of Temporal Database Concepts. *SIGMOD record*, 23:52–64, 1994.

[Jensen *et al.*, 1996] C.S. Jensen, R.T. Snodgrass, and M.D. Soo. Extending Existing Dependency Theory to Temporal Databases. *IEEE Transactions on Knowledge and Data Engineering*, 8(4), 1996.

[Jensen, 1995] C. S. Jensen. Vacuuming. In R. T. Snodgrass, editor, *The TSQL2 Temporal Query Language*, pages 447–460. Kluwer Academic Publishers, 1995.

[Jensen, 2001] F.V. Jensen. *Bayesian Networks and Decision Graphs*. Springer-Verlag, New York, 2001.

[Johnson *et al.*, 2000] P. Johnson, S. Tu, N. Booth, B. Sugden, and I. Purves. Using Scenarios in Chronic Disease Management Guidelines for Primary Care. In M.J. Overhage, editor, *Proceedings of the AMIA Annual Symposium*. Hanley & Belfus, 2000.

[Jonsson and Bäckström, 1996] P. Jonsson and C. Bäckström. A Linear Programming Approach to Temporal Reasoning. In *Proceedings of the AAAI Conference*, pages 1235–1240. AAAI Press/MIT Press, 1996.

[Jonsson and Bäckström, 1998] Peter Jonsson and Christer Bäckström. A unifying approach to temporal constraint reasoning. *Artificial Intelligence*, 102(1):143–155, 1998.

[Jonsson *et al.*, 1999] Peter Jonsson, Thomas Drakengren, and Christer Bäckström. Computational complexity of relating time points with intervals. *Artificial Intelligence*, 109(1–2):273–295, 1999.

[Josephson and Josephson, 1994] J.R. Josephson and S.G. Josephson, editors. *Abductive Inference: Computation, Philosophy, Technology*. New York: Cambridge University Press, 1994.

[Jung *et al.*, 1996] C.G. Jung, K. Fischer, and A. Burt. Multi-Agent Planning Using an Abductive Event Calculus. Technical Report DFKI Report RR-96-04, DFKI, Germany, 1996.

[Kabanza *et al.*, 1995] F. Kabanza, J-M. Stevenne, and P. Wolper. Handling Infinite Temporal Data. *Journal of Computer and System Sciences*, 51(1):3–17, 1995.

[Kahn and Gorry, 1977] K. Kahn and G.A. Gorry. Mechanizing Temporal Knowledge. *Artificial Intelligence*, 9:87–108, 1977.

[Kahn, 1988] M.G. Kahn. *Model-Based Interpretation of Time-Ordered Medical Data*. PhD thesis, Section on Medical Information Sciences, University of California, San Francisco, USA, 1988.

[Kahn, 1991a] M.G. Kahn. Combining Physiologic Models and Symbolic Methods to Interpret Time-Varying Patient Data. *Methods of Information in Medicine*, 30:167–178, 1991.

[Kahn, 1991b] M.G. Kahn. Extensions to the Time-Oriented Database Model to Support Temporal Reasoning in Medical Expert Systems. *Methods of Information in Medicine*, 30:4–14, 1991.

[Kahn, 1991c] M.G. Kahn. TQuery: A Context-Sensitive Temporal Query Language. *Computers and Biomedical Research*, 24:401–419, 1991.

[Kaivola, 1995] R. Kaivola. Axiomatising Linear Time mu-Calculus. In *Proceedings of the Sixth International Conference on Concurrency theory (CONCUR)*, volume 962 of *LNCS*, pages 423–437. Springer-Verlag, 1995.

[Kakas and Mancarella, 1990a] A. Kakas and P. Mancarella. Generalized Stable Models: A Semantics for Abduction. In *Proceedings of European Conference on Artificial Intelligence (ECAI)*, pages 385–391, 1990.

[Kakas and Mancarella, 1990b] A.C. Kakas and P. Mancarella. Database Updates Through Abduction. In *Proceedings of the Sixteenth International Conference on Very Large Databases (VLDB)*, pages 650–661. Morgan Kaufmann, 1990.

[Kakas and Miller, 1997] A. Kakas and R. Miller. A Simple Declarative Language for Describing Narratives with Actions. *Journal of Logic Programming*, 31(1-3):157–200, April-June 1997.

[Kakas *et al.*, 1992] A. C. Kakas, R.A. Kowalski, and F. Toni. Abductive Logic Programming. *Journal of Logic and Computation*, 2(6):719–770, 1992.

[Kakas *et al.*, 2000] A.C. Kakas, A. Michael, and C. Mourlas. ACLP: Abductive Constraint Logic Programming. *Journal of Logic Programming*, 44(1-3):129–177, 2000.

[Kakas *et al.*, 2001] A. Kakas, B. Van Nuffelen, and M. Denecker. A-system : Problem Solving Through Abduction. In B. Nebel, editor, *Proceedings of Seventeenth International Joint Conference on Artificial Intelligence (IJCAI)*, volume 1, pages 591–596. Morgan Kaufmann Publishers, Inc., 2001.

[Kamp and Reyle, 1993] H. Kamp and U. Reyle. *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht, 1993.

[Kamp and Reyle, 1996] H. Kamp and U. Reyle. A Calculus for First-Order Discourse Representation Structures. *Journal of Logic, Language and Information*, 5:297–348, 1996.

[Kamp, 1968] H. Kamp. *Tense Logic and the Theory of Linear Order*. PhD Dissertation, UCLA, Los Angeles, USA, 1968.

[Kamp, 1971] H. Kamp. Formal Properties of 'now'. *Theoria*, 37:227–273, 1971.

[Kamp, 1979] H. Kamp. Events, Instants and Temporal Reference. In R. Bäuerle, U. Egli, and A. von Stechow, editors, *Semantics from Different Points of View*, pages 376–417. Springer-Verlag, 1979.

[Kanazawa *et al.*, 1995] K. Kanazawa, D. Koller, and S. Russell. Stochastic Simulation Algorithms for Dynamic Probabilistic Networks. In *Proceedings of International Conference on Uncertainty in AI (UAI)*, 1995.

[Kanazawa, 1991] K. Kanazawa. A Logic and Time Nets for Probabilistic Inference. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI)*, pages 360–365. MIT Press, 12–19 July 1991.

[Kanellakis *et al.*, 1990] P.C. Kanellakis, G.M. Kuper, and P.Z. Revesz. Constraint Query Languages. In *Proceedings of the Ninth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 299–313, 1990.

[Kanellakis *et al.*, 1995] P.C. Kanellakis, G.M. Kuper, and P.Z. Revesz. Constraint Query Languages. *Journal of Computer and System Sciences*, 51(1):26–52, August 1995.

[Karmarkar, 1984] N. Karmarkar. A New Polynomial Time Algorithm for Linear Programming. *Combinatorica*, 4:373–395, 1984.

[Kartha, 1993] G. Kartha. Soundness and Completeness Theorems for Three Formalizations of Action. In *Proceedings if International joint Conference on Artificial Intelligence (IJCAI)*, pages 724–729, 1993.

[Kautz and Ladkin, 1991] H. Kautz and P. Ladkin. Integrating Metric and Temporal Qualitative Temporal Reasoning. In *Proceedings of the Ninth (US) National Conference on Artificial Intelligence (AAAI)*, pages 241–246, Anaheim, USA, July 1991. American Association for Artificial Intelligence, AAAI Press/MIT Press.

[Kautz and Selman, 1995] H. Kautz and B. Selman. Unifying SAT-based and Graph-based Planning. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 318–325. Morgan Kaufmann, 1995.

[Kautz, 1987] H.A. Kautz. *A Formal Theory of Plan Recognition*. PhD thesis, Department of Computer Science, University of Rochester, Rochester, USA, May 1987. Available as Technical Report 215.

[Kautz, 1991] H.A. Kautz. A Formal Theory of Plan Recognition and its Implementation. In *Reasoning about Plans*, pages 69–126. Morgan Kaufmann, San Mateo, CA, 1991.

[Kay *et al.*, 2000] H. Kay, B. Rinner, and B. J. Kuipers. Semi-quantitative System Identification. *Artificial Intelligence*, 119:103–140, 2000.

[Keenan and Westerstahl, 1997] E. Keenan and D. Westerstahl. Generalized Quantifiers in Linguistics and Logic. In J. van Benthem and A. ter Meulen, editors, *Handbook of Logic and Language*, pages 837–893. MIT/Elsevier, 1997.

[Kenny, 1963] A. Kenny. *Action, Emotion, and Will*. Routledge and Kegan Paul, London, 1963.

[Keravnou and Washbrook, 1990] E.T. Keravnou and J. Washbrook. A Temporal Reasoning Framework used in the Diagnosis of Skeletal Dysplasias. *Artificial Intelligence in Medicine*, 2:239–265, 1990.

[Keravnou and Washbrook, 2001] E.T. Keravnou and J. Washbrook. Abductive Diagnosis using Time-Objects: Criteria for the Evaluation of Solutions. *Computational Intelligence*, 17:87–131, 2001.

[Keravnou, 1996a] E.T. Keravnou. An Ontology of Time Using Time-Axes and Time-Objects as Primitives. Technical Report TR-96-9, Department of Computer Science, University of Cyprus, Cyprus, 1996.

[Keravnou, 1996b] E.T. Keravnou. Temporal Diagnostic Reasoning based on Time-Objects. *Artificial Intelligence in Medicine*, 8:235–265, 1996.

[Keravnou, 1997] E.T. Keravnou. Temporal Abstraction of Medical Data: Deriving Periodicity. In N. Lavrač, E.T. Keravnou, and B. Zupan, editors, *Intelligent Data Analysis in Medicine and Pharmacology*, pages 61–79. Kluwer Academic Publishers, 1997.

[Keravnou, 1999] E.T. Keravnou. A Multidimensional and Multigranular Model of Time for Medical Knowledge–Based Systems. *Journal of Intelligent Information Systems*, 13:79–120, 1999.

[Kesten and Pnueli, 1995] Y. Kesten and A. Pnueli. A Complete Proof System for QPTL. In *Logics of Programs*, pages 2–12. IEEE, 1995.

[Kesten *et al.*, 1994] Y. Kesten, Z. Manna, and A. Pnueli. Temporal Verification of Simulation and Refinement. *Lecture Notes in Computer Science*, 803, 1994.

[Kesten *et al.*, 1997] Y. Kesten, Z. Manna, H. McGuire, and A. Pnueli. A Decision Algorithm for Full Propositional Temporal Logic. In *Proceedings of Conference on Computer Aided Verification (CAV)*, volume 697 of *Lecture Notes in Computer Science*, pages 97–109. Springer, 1997.

[Khachiyan, 1979] L. G. Khachiyan. A Polynomial Algorithm in Linear Programming. *Soviet Mathematics Doklady*, 20:191–194, 1979.

[Kifer and Lausen, 1989] M. Kifer and G. Lausen. F-logic: A Higher-order Language for Reasoning about Objects. In *Proceedings of Eighth ACM-SIGACT-SIGMOD-SIGART Symposium of Principles of Database Systems (PODS)*, 1989.

[Kjaerulff, 1994] U. Kjaerulff. A Computational Scheme for Reasoning in Dynamic Probabilistic Networks. In *Proceedings of International Conference on Uncertainty in AI (UAI)*, pages 121–129, 1994.

[Kleene, 1956] S. Kleene. Representation of Events in Nerve Nets and Finite Automata. In C. Shannon and J. McCarthy, editors, *Automata Studies*, pages 3–41. Princeton Univ. Press, 1956.

[Kline, 1993] N. Kline. An Update of the Temporal Database Bibliography. *SIGMOD RECORD*, 22(4):66–80, 1993.

[Konev *et al.*, 2003] B. Konev, A. Degtyarev, C. Dixon, M. Fisher, and U. Hustadt. Towards the Implementation of First-Order Temporal Resolution: the Expanding Domain Case. In *Proceedings of the Joint Tenth International Symposium on Temporal Representation and Reasoning and Fourth International Conference on Temporal Logic (TIME-ICTL)*. IEEE Press, 2003.

[Konev, 2003] B. Konev. TRP++: Temporal Resolution Prover, 2003. Department of Computer Science, University of Liverpool, UK. `http://www.csc.liv.ac.uk/~konev/trp++`.

[Kong *et al.*, 1994] A. Kong, J. Liu, and W.H. Wong. Sequential Imputations and Bayesian Missing Data Problems. *Journal of the American Statistical Association*, 89:278–298, 1994.

[Kontchakov *et al.*, 2003] R. Kontchakov, C. Lutz, F. Wolter, and M. Zakharyaschev. Temporalizing Tableaux. *Studia Logica*, 2003.

[Koubarakis and Skiadopoulos, 1999] M. Koubarakis and S. Skiadopoulos. Querying Temporal Constraint Networks in PTIME. In *Proceedings of AAAI Conference*, pages 745–750, 1999.

[Koubarakis and Skiadopoulos, 2000] M. Koubarakis and S. Skiadopoulos. Querying Temporal and Spatial Constraint Networks in PTIME. *Artificial Intelligence*, 123(1-2):223–263, 2000.

[Koubarakis, 1992] M. Koubarakis. Dense Time and Temporal Constraints with $\neq$. In Swartout and Nebel [1992], pages 24–35.

[Koubarakis, 1993] M. Koubarakis. Representation and Querying in Temporal Databases: the Power of Temporal Constraints. In *Proceedings of the Ninth International Conference on Data Engineering*, pages 327–334, April 1993.

[Koubarakis, 1994a] M. Koubarakis. Complexity Results for First-Order Theories of Temporal Constraints. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 379–390. Morgan Kaufmann, San Francisco, CA, May 1994.

[Koubarakis, 1994b] M. Koubarakis. Database Models for Infinite and Indefinite Temporal Information. *Information Systems*, 19(2):141–173, March 1994.

[Koubarakis, 1995] M. Koubarakis. From Local to Global Consistency in Temporal Constraint Networks. In *Proceedings of the First International Conference on Principles and Practice of Constraint Programming (CP)*, volume 976 of *Lecture Notes in Computer Science*, pages 53–69, Cassis, France, September 1995.

[Koubarakis, 1996] M. Koubarakis. Tractable Disjunctions of Linear Constraints. In *Proceedings of the Second International Conference on Principles and Practice of Constraint Programming (CP)*, pages 297–307, Cambridge, MA, USA, August 1996.

[Koubarakis, 1997a] M. Koubarakis. From Local to Global Consistency in Temporal Constraint Networks. *Theoretical Computer Science*, 173:89–112, February 1997.

[Koubarakis, 1997b] M. Koubarakis. The Complexity of Query Evaluation in Indefinite Temporal Constraint Databases. *Theoretical Computer Science*, 171:25–60, January 1997.

[Koubarakis, 2001] M. Koubarakis. Tractable Disjunctions of Linear Constraints: Basic Results and Applications to Temporal Reasoning. *Theoretical Computer Science*, 266:311–339, September 2001.

[Kowalski and Sadri, 1997] R. Kowalski and F. Sadri. Reconciling the Event Calculus with the Situation Calculus. *Journal of Logic Programming*, 31(1-3):39–58, 1997.

[Kowalski and Sergot, 1986] R.A Kowalski and M.J. Sergot. A Logic-based Calculus of Events. *New Generation Computing*, 1(4):67–95, 1986.

[Kowalski, 1992] R.A. Kowalski. Database Updates in the Event Calculus. *Journal of Logic Programming*, 1992.

[Koymans and Roever, 1985] R. Koymans and W.-P. de Roever. Examples of a Real-Time Temporal Logic Specification. In B.T. Denvir, W.T. Harwood, M.I. Jackson, and M.J. Wray, editors, *Analysis of Concurrent Systems*, volume 207 of *Lecture Notes in Computer Science*, pages 231–251. Springer-Verlag, Berlin-Heidelberg-New York, 1985.

[Koymans, 1989] R. Koymans. *Specifying Message Passing and Time-Critical Systems with Temporal Logic*. PhD thesis, Technische Universiteit Eindhoven, Netherlands, 1989.

[Kozen, 1982] D. Kozen. Results on the Propositional μ-calculus. In *Proceeings of ICALP*, number 140 in Lecture Notes in Computer Science, pages 340–359, 1982.

[Kroger, 1987] F. Kroger. *Temporal Logic of Programs*. Springer Verlag, 1987.

[Krokhin *et al.*, 2001] A. Krokhin, P. Jeavons, and P. Jonsson. A Complete Classification of Tractability in Allen's Algebra in the Presence of a Non-Trivial Basic Relation. In B. Nebel, editor, *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 83–88, Seattle, Washington, USA, August 2001. Morgan Kaufmann.

[Krokhin *et al.*, 2003] A. Krokhin, P. Jeavons, and P. Jonsson. The Tractable Subalgebras of Allen's Interval Algebra. *Journal of the ACM*, 50(5):591–640, 2003.

[Kuipers and Åström, 1994] B. J. Kuipers and K. Åström. The Composition and Validation of Heterogeneous Control Laws. *Automatica*, 30(2):233–249, 1994.

[Kuipers, 1986] B. Kuipers. Qualitative Simulation. *Artificial Intelligence*, 26:289–338, 1986.

[Kuipers, 1988] B. Kuipers. Qualitative Simulation using Time-Scaled Abstraction. *Artificial Intelligence in Engineering*, 3(4):185–191, 1988.

[Kuipers, 1994] B. J. Kuipers. *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*. MIT Press, Cambridge, MA, 1994.

[Kumari and Pujari, 2002] G.V. Kumari and K. Pujari. Enforcing the Local Consistency in INDU. In *Proceedings of the International Conference on Knowledge Based Computed Systems*, 2002.

[Kuper, 1987] G. Kuper. Logic Programming with Sets. In *Proceedings of the Sixth ACM-SIGACT-SIGMOD-SIGART Symposium of Principles of Database Systems (PODS)*, 1987.

[Kushmerick *et al.*, 1995] N. Kushmerick, S. Hanks, and D. Weld. An Algorithm for Probabilistic Planning. *Artificial Intelligence*, 76:239–286, 1995.

[Kvarnström *et al.*, 2000] J. Kvarnström, P. Doherty, and P. Hasslum. Extending TALplanner with Concurrency and Resources. In *Proceedings of the Europena Conference on Artificila Intelligence (ECAI)*, Berlin, Germany, August 2000.

[Ladkin and Maddux, 1988] P.B. Ladkin and R. Maddux. On Binary Constraint Networks. Technical Report KES.U.88.8, Kestrel Institute, Palo Alto, USA, 1988.

[Ladkin and Maddux, 1994] P.B. Ladkin and R. Maddux. On Binary Constraint Problems. *Journal of the ACM*, 41(3):435–469, 1994.

[Ladkin and Reinefeld, 1992] P.B. Ladkin and A. Reinefeld. Effective Solution of Qualitative Interval Constraint Problems. *Artificial Intelligence*, 57(1):105–124, 1992.

[Ladkin and Reinefeld, 1997] P.B. Ladkin and A. Reinefeld. Fast Algebraic Methods for Interval Constraint Problems. *Annals of Mathematics and Artificial Intelligence*, 19:383–411, 1997.

[Ladkin, 1987] P. Ladkin. The Completeness of a Natural System for Reasoning with Time Intervals. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 462–467, 1987.

[Ladkin, 1988] P. Ladkin. Satisfying First-Order Constraints About Time Intervals. In *Proceedings of AAAI Conference*, pages 512–517, 1988.

[Lansky, 1986] A.L. Lansky. A Representation of Parallel Activity Based on Events, Structure and Causality. In M.P. Georgeff and A.L. Lansky, editors, *Proceedings of the Timberline Oregon Workshop on Reasoning about Actions and Plans*, 1986.

[Larizza *et al.*, 1992] C. Larizza, A. Moglia, and M. Stefanelli. M-HTP: A System for Monitoring Heart-Transplant Patients. *Artificial Intelligence in Medicine*, 4(2):111–126, 1992.

[Larizza *et al.*, 1997] C. Larizza, R. Bellazzi, and A. Riva. Temporal Abstractions for Diabetic Patients Management. In *Proceedings of AIME Conference*, volume 1211 of *Lecture Notes in Artificial Intelligence*, pages 319–330. Springer, 1997.

[Lascarides and Asher, 1993] A. Lascarides and N. Asher. Temporal Interpretation, Discourse Relations and Commonsense Entailment. *Linguistics and Philosophy*, 16:437–594, 1993.

[Lascarides and Oberlander, 1993] A. Lascarides and J. Oberlander. Temporal Connectives in a Discourse Context. In *Proceedings the Sixth Conference of the European Chapter of the ACL (EACL)*, pages 260–268, 1993.

[Last *et al.*, 2004] M. Last, A. Kandel, and H. Bunke, editors. *Data Mining in Time Series Databases*, volume 57 of *Machine Perception and Artificial Intelligence*. World Scientific, 2004.

[Lausen *et al.*, 1998] G. Lausen, B. Ludäscher, and W. May. On Logical Foundations of Active Databases. In J. Chomicki and G. Saake, editors, *Logics for Databases and Information Systems*, pages 389–422. Kluwer, 1998.

[Lavrač *et al.*, 1997] N. Lavrač, E.T. Keravnou, and B. Zupan, editors. *Intelligent Data Analysis in Medicine and Pharmacology*. Kluwer Academic Publishers, 1997.

[Leban *et al.*, 1986] B. Leban, D. Mcdonald, and D. Foster. A Representation for Collections of Temporal Intervals. In *Proceedings of the Fifth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 367–371, 1986.

[Lenzerini, 2002] M. Lenzerini. Data Integration: a Theoretical Perspective. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 233–246, 2002.

[Levesque, 1984] H.J. Levesque. Foundations of a Functional Approach to Knowledge Representation. *Artificial Intelligence*, 23:155–212, 1984.

[Levy *et al.*, 1995] A. Y. Levy, A. O. Mendelzon, Y. Sagiv, and D. Srivastava. Answering Queries Using Views. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 95–104, 1995.

[Libkin *et al.*, 2000] L. Libkin, G. Kuper, and J. Paredaens, editors. *Constraint Databases*. Springer, 2000.

[Lichtenstein and Pnueli, 1985] O. Lichtenstein and A. Pnueli. Checking that Finite State Concurrent Programs Satisfy their Linear Specification. In *Proceedings of the Twelfth ACM Symposium on the Principles of Programming Languages (POPL)*, pages 97–107, New Orleans, USA, January 1985.

[Lichtenstein *et al.*, 1985] O. Lichtenstein, A. Pnueli, and L. Zuck. The Glory of the Past. In *Logics of Programs*, volume 193 of *Lecture Notes in Computer Science*, pages 196–218. Springer-Verlag, Heidelberg, 1985.

[Lifschitz and Turner, 1994] V. Lifschitz and H. Turner. Splitting a Logic Program. In P. Van Hentenryck, editor, *Proceedings of the Eleventh International Conference on Logic Programming (ICLP)*, pages 23–38, 1994.

[Lifschitz, 1987] V. Lifschitz. Formal Theories of Action. In *Proceedings of the Workshop on the Frame Problem*, pages 35–37, 1987.

[Lifschitz, 1997] V. Lifschitz. On the Logic of Causal Explanation. *Artificial Intelligence*, 96:451–465, 1997.

[Lifschitz, 1999] V. Lifschitz. Answer Set Planning. In *Proceedings of International Conference on Logic Programming (ICLP)*, pages 23–37, 1999.

[Ligozat, 1990] G. Ligozat. Weak Representations of Interval Algebras. In *Proceedings of AAAI Conference*, pages 715–720, 1990.

[Ligozat, 1996] G. Ligozat. A New Proof of Tractability for ORD-Horn Relations. In *Proceedings of the Thirteenth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 715–720. AAAI Press/The MIT Press, 1996.

[Lin, 1991] Y. Lin. Two Theories of Time. *Journal of Applied Non-Classical Logics*, 1(1):37–63, 1991.

[Lin, 1995] F. Lin. Embracing Causality in Specifying the Indirect Effects of Actions. In *Proceedings of the Internationl Joint Conference on Artificial Intelligence (IJCAI)*, pages 1985–1993, 1995.

[Lin, 1997] F. Lin. Application of the Situation Calculus to Formalizing Control and Strategic Information: the Prolog Cut Operator. In *Proceedings of the Internationl Joint Conference on Artificial Intelligence (IJCAI)*, pages 1412–1418, 1997.

[Lipski, 1979] W. Lipski. On Semantic Issues Connected with Incomplete Information Databases. *ACM Transcactions on Database Systems*, 4(3):262–296, September 1979.

[Littman, 1997] M. L. Littman. Probabilistic proposition planning: Representations and complexity. In *Proceedings of AAAI Conference*, 1997.

[Liu and Chen, 1998] J. Liu and R. Chen. Sequential Monte Carlo Methods for Dynamic Systems. *Journal of the American Statistical Association*, 93, 1998.

[Liu and Orgun, 1997] C. Liu and M. Orgun. Dealing with Multiple Granularity of Time in Temporal Logic Programming. *Journal of Symbolic Computation*, 22(5):699–720, 1997.

[Lloyd and Topor, 1984] J.W. Lloyd and R.W. Topor. Making Prolog More Expressive. *Journal of Logic Programming*, 1(3):225–240, 1984.

[Lloyd, 1987] J.W. Lloyd. *Foundations of Logic Programming*. Symbolic Computation Series. Springer-Verlag, 2 edition, 1987.

[Lobo *et al.*, 1992] J. Lobo, J. Minker, and A. Rajasekar. *Foundations of Disjunctive Logic Programming*. The MIT Press, 1992.

[Long and Fox, 2003a] D. Long and M. Fox. Exploiting a Graphplan Framework in Temporal Planning. In *Proceedings of International Conference on Automated Planning and Scheduling (ICAPS)*, 2003.

[Long and Fox, 2003b] D. Long and M. Fox. The 3rd International Planning Competition: Results and Analysis. *Journal of AI Research*, 20, 2003.

[Long, 1996] W.J. Long. Temporal Reasoning for Diagnosis in a Causal Probabilistic Knowledge Base. *Artificial Intelligence in Medicine*, 8:193–215, 1996.

[Lorentzos and Mitsopoulos, 1997] N. A. Lorentzos and Y. G. Mitsopoulos. SQL Extension for Interval Data. *IEEE Transactions on Knowledge and Data Engineering*, 9(3):480–499, 1997.

[Lorentzos *et al.*, 1995] N. A. Lorentzos, A. Poulovassilis, and C. Small. Manipulation Operations for an Interval-extended Relational Model. *Data and Knowledge Engineering*, 17(1):1–29, 1995.

[Lorentzos, 1993] N. A. Lorentzos. The Interval-extended Relational Model and Its Application to Valid-time Databases. In Tansel et al. [1993], pages 67–91.

[Lutz, 1999a] C. Lutz. Complexity of Terminological Reasoning Revisited. In *Proceedings of the Sixth International Conference on Logic for Programming and Automated Reasoning (LPAR)*, pages 181–200. Springer-Verlag, 6 – 10, 1999.

[Lutz, 1999b] C. Lutz. Reasoning with Concrete Domains. In T. Dean, editor, *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 90–95, Stockholm, Sweden, 31 – 6, 1999. Morgan-Kaufmann Publishers.

[Mackaay *et al.*, 1990] E. Mackaay, D. Poulin, J. Fremont, P. Bratley, and C. Deniger. The Logic of Time in Law and Legal Expert Systems. *Ratio Juris*, 3(2):254–271, 1990.

[Mackworth, 1977] A.K. Mackworth. Consistency in Networks of Relations. *Artificial Intelligence*, 8(1):99–118, 1977.

[Madden *et al.*, 2002] S. Madden, M. A. Shah, J. M. Hellerstein, and V. Raman. Continuously Adaptive Continuous Queries over Streams. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 49–60, 2002.

[Maier, 1986] D. Maier. A Logic for Objects. In *Proceedings of the Workshop on Foundations of Deductive Database and Logic Programming*, 1986.

[Majercik and Littman, 1999] S. Majercik and M. Littman. Contingent Planning under Uncertainty via Stochastic Satisfiability. In *Proceedings of Sixteenth National Conference on AI*, 1999.

[Manna and Pnueli, 1992] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems*. Springer-Verlag, Berlin, Heidelberg, New York, 1992.

[Manna and Pnueli, 1995] Z. Manna and A. Pnueli. *Temporal Verification of Reactive Systems: Safety*. Springer-Verlag, New York, 1995.

[Manna and Wolper, 1984] Z. Manna and P. Wolper. Synthesis of Communicating Processes from Temporal Logic Specifications. *ACM Transactions on Programming Languages and Systems*, 6(1):68–93, January 1984.

[Manzano, 1993] M. Manzano. *Many-Sorted Logic and its Applications in Computer Science*, chapter 1 (Introduction to Many-sorted logic; pp. 1-86). John Wiley&Sons. Chichester. (UK), tucker & meinke edition, 1993.

[Marek and Truszczyński, 1994] W. Marek and M. Truszczyński. Revision Programming, Database Updates and Integrity Constraints. In *Proceedings of Fifth International Conference in Database Theory (ICDT)*, Prague, Czechoslovakia, 1994.

[Marek and Truszczyński, 1999] W. Marek and M Truszczyński. Stable Models and an Alternative Logic Programming Paradigm. In Apt, K. and Marek, W. and Truszczyński, M. and Warren, D., editor, *The Logic Programming Paradigm: a 25-Year perspective*, pages 375–398. Springer, 1999.

[Maruichi *et al.*, 1991] T. Maruichi, M. Ichikawa, and M. Tokoro. Modelling Autonomous Agents and their Groups. In Y. Demazeau and J. P. Müller, editors, *Decentralized AI 2 – Proceedings of the 2^{nd} European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW)*. Elsevier/North Holland, 1991.

[McAllester and Rosenblitt, 1991] D. McAllester and D. Rosenblitt. Systematic Nonlinear Planning. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI)*, volume 2, pages 634–639, Anaheim, California, USA, 1991. AAAI Press/MIT Press.

[McCain and Turner, 1994] N. McCain and H. Turner. Language Independence and Language Tolerance in Logic Programs. In *Proceedings of the Eleventh International Conference on Logic Programming (ICLP)*, pages 38–57, 1994.

[McCain and Turner, 1995] N. McCain and H. Turner. A causal theory of ramifications and qualifications. In *Proc. of IJCAI 95*, pages 1978–1984, 1995.

[McCain and Turner, 1997] N. McCain and H. Turner. Causal Theories of Action and Change. In H. Shrobe and T. Senator, editors, *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference (AAAI)*, pages 460–465, Menlo Park, California, 1997. AAAI Press.

[McCain and Turner, 1998] N. McCain and H. Turner. Satisfiability Planning with Causal Theories. In *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 212–223, 1998.

[McCarthy and Hayes, 1969] J. McCarthy and P. J. Hayes. Some Philosophical Problems from the Standpoint of Artificial Intelligence. In B. Melzer and D. Michie, editors, *Machine Intelligence 4*. Edinburgh University Press, 1969.

[McCarthy, 1959] J. McCarthy. Programs with Common Sense. In *Proceedings of the Teddington Conference on the Mechanization of Thought Processes*, pages 75–91, London, 1959. Her Majesty's Stationery Office.

[McCarthy, 1963] J. McCarthy. Situations, Actions and Causal Laws. Technical Report Memo 2, Stanford Artificial Intelligence Project, 1963.

[McCarthy, 1980] J. McCarthy. Circumscription – A Form of Non-Monotonic Reasoning. *Artificial Intelligence*, 13:27–39, 1980.

[McCarty, 1995] L. Thorne McCarty. Some requirements on an action language for legal discourse (position paper). In *Spring Symposium Series'95: Extending Theories of Action*, pages 136–138. AAAI, 1995.

[McDermott and Doyle, 1980] D. McDermott and J. Doyle. Non-Monotonic Logic I. *Artificial Intelligence*, 13:41–72, 1980.

[McDermott, 1996] D. McDermott. A Heuristic Estimator for Means Ends Analysis in Planning. In B. Drabble, editor, *Proceedings of the Third International Conference on Artificial Intelligence Planning Systems (AIPS)*, pages 142–149. AAAI Press, 1996.

[McDermott, 2000] D. McDermott. The 1998 AI Planning Systems Competition. *AI Magazine*, 21(2), 2000.

[McDermott, 2003] D. McDermott. Reasoning about Autonomous Processes in an Estimated-Regression Planner. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2003.

[McGuire, 1995] H. W. McGuire. *Two Methods for Checking Formulas of Temporal Logic*. PhD thesis, Department of Computer Science, Stanford University, USA, June 1995. Stanford Computer Science Technical Reports CS-TR-95-1551.

[McNaughton, 1966] R. McNaughton. Testing and Generating Infinite Sequences by Finite Automata. *Information and Control*, 9:521–530, 1966.

[Meiri, 1991] I. Meiri. Combining Qualitative and Quantitative Constraints in Temporal Reasoning. In *Proceedings of AAAI Conference*, pages 260–267, 1991.

[Meiri, 1996] I. Meiri. Combining Qualitative and Quantitative Constraints in Temporal Reasoning. *Artificial Intelligence*, 87(1–2):343–385, 1996.

[Meyer *et al.*, 1999] J.-J. Ch. Meyer, W. van der Hoek, and B. van Linder. A Logical Approach to the Dynamics of Commitments. *Artificial Intelligence*, 113:1–40, 1999.

[Miksch *et al.*, 1996] S. Miksch, W. Horn, C. Popow, and F. Paky. Utilizing Temporal Data Abstraction for Data Validation and Therapy Planning for Artificially Ventilated Newborn Infants. *Artificial Intelligence in Medicine*, 8(6):543–576, 1996.

[Miller and Schubert, 1990] S. A. Miller and L.K. Schubert. Time Revisited. *Computational Intelligence*, 6:108–118, 1990.

[Miller and Shanahan, 1994] R. Miller and M. Shanahan. Narratives in the Situational Calculus. *Journal of Logic and Computation*, 4(5):513–530, 1994.

[Miller *et al.*, 1982] R.A. Miller, H.E. Pople, and J.D. Myers. INTERNIST-I, An Experimental Computer-Based Diagnostic Consultant for General Internal Medicine. *New England Journal of Medicine*, 307:468–476, 1982.

[Miller, 1986] P. L. Miller. *Expert Critiquing Systems: Practice-Based Medical Consultation by Computer*. Springer-Verlag, New York, NY, 1986.

[Miller, 1990] B. Miller. The Rhetorical Knowledge Representation System Reference Manual. Technical Report 326, Department of Computer Science, University of Rochester, Rochester, New York, USA, 1990.

[Missiaen *et al.*, 1992] L. R. Missiaen, M. Bruynooghe, and M. Denecker. Abductive Planning with Event Calculus. Internal report, Department of Computer Science, K.U.Leuven, 1992.

[Missiaen *et al.*, 1995] L. R. Missiaen, M. Denecker, and M. Bruynooghe. CHICA, An Abductive Planning System Based on Event Calculus. *Journal of Logic and Computation*, 5(5):579–602, September 1995.

[Missiaen, 1991a] L. R. Missiaen. Localized Abductive Planning for Robot Assembly. In *Proceedings of the IEEE Conference on Robotics and Automation*, pages 605–610. IEEE Robotics and Automation Society, 1991.

[Missiaen, 1991b] L. R. Missiaen. *Localized Abductive Planning with the Event Calculus*. PhD thesis, Department of Computer Science, K.U.Leuven, 1991.

[Miyano and Hayashi, 1984] S. Miyano and T. Hayashi. Alternating Finite Automata on ω-words. *Theoretical Computer Science*, 32:321–330, 1984.

[Mokhtar *et al.*, 2002] H. Mokhtar, J. Su, and O. Ibarra. On Moving Objects Queries. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, pages 188–198, 2002.

[Montanari and de Rijke, 1997] A. Montanari and M. de Rijke. Two-Sorted Metric Temporal Logic. *Theoretical Computer Science*, 183:187–214, 1997.

[Montanari and Policriti, 1996] A. Montanari and A. Policriti. Decidability Results for Metric and Layered Temporal Logics. *Notre-Dame Journal of Formal Logic*, 37:260–282, 1996.

[Montanari and Puppis, 2004a] A. Montanari and G. Puppis. Decidability of MSO theories of tree structures. In *Proceedings of the Twenty-Fourth Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, Lecture Notes in Computer Science. Springer-Verlag, 2004.

[Montanari and Puppis, 2004b] A. Montanari and G. Puppis. Decidability of the Theory of the Totally Unbounded ω-Layered Structure. In *Proceedings of the Eleventh International Symposium on Temporal Representation and Reasoning (TIME)*, pages 156–160. IEEE Computer Society Press, 2004.

[Montanari *et al.*, 1992] A. Montanari, E. Ciapessoni, E. Corsetti, and P. San Pietro. Dealing with Time Granularity in Logical Specifications of Real-Time Systems: The Synchronous Case. Technical Report 7, Dipartimento di Matematica ed Informatica, Universitá di Udine, Udine (IT), May 1992.

[Montanari *et al.*, 1999] A. Montanari, A. Peron, and A. Policriti. Theories of Omega-Layered Metric Temporal Structures: Expressiveness and Decidability. *Logic Journal of the IGPL*, 7(1):79–102, 1999.

[Montanari *et al.*, 2000] A. Montanari, A. Peron, and A. Policriti. The Taming (Timing) of the States. *Logic Journal of the IGPL*, 8(5):681–699, 2000.

[Montanari *et al.*, 2002a] A. Montanari, A. Peron, and A. Policriti. Extending Kamp's Theorem to Model Time Granularity. *Journal of Logic and Computation*, 12(4):641–678, 2002.

[Montanari *et al.*, 2002b] A. Montanari, G. Sciavicco, and N. Vitacolonna. Decidability of Interval Temporal Logics over Split-Frames via Granularity. In *Proceedings of the European Conference on Logic in Artificial Intelligence (JELIA)*, number 2424 in Lecture Notes in Artificial Intelligence, pages 259–270. Springer, 2002.

[Montanari, 1974] U. Montanari. Networks of Constraints: Fundamental Properties and Applications to Picture Processing. *Information Science*, 7(3):95–132, 1974.

[Montanari, 1994] A. Montanari. A Layered and Metric Temporal Logic for Time Granularity, Synchrony and Asynchrony. Technical Report MPI-I-94-230, Max-Plank-Institut fuer Informatik, July 1994.

[Montanari, 1996] A. Montanari. *Metric and Layered Temporal Logic for Time Granularity*. PhD thesis, University of Amsterdam, Amsterdam, Netherlands, september 1996. ILLC Dissertation Series 1996-02.

[Monti and Peron, 2000] A. Monti and A. Peron. Systolic Tree ω-Languages: The Operational and the Logical View. *Theoretical Computer Science*, 23:1–17, 2000.

[Monti and Peron, 2001] A. Monti and A. Peron. Logical Definability of Y-Tree and Trellis Systolic ω-Languages. *Acta Cybernetica*, 15:75–100, 2001.

[Moore, 1990] R. C. Moore. A Formal Theory of Knowledge and Action. In J. F. Allen, J. Hendler, and A. Tate, editors, *Readings in Planning*, pages 480–519. Morgan Kaufmann, 1990.

[Morris *et al.*, 1997] R. Morris, W. Shoaf, and L. Khatib. Domain Independent Temporal Reasoning with Recurring Events. *Computational Intelligence*, 12(3):450–477, 1997.

[Morurovic *et al.*, 2000] M. Morurovic, F. Wolter, and M. Zakharyaschev. Modalized Description Logic — How Much? Technical report, Computer Science Department, University of Leipzig, Germany, 2000.

[Moszkowski, 1983] B. Moszkowski. *Reasoning about Digital Circuits*. PhD thesis, Department of Computer Science, Stanford University, Technical Report STAN-CS-83-970, Stanford, CA, USA, 1983.

[Moszkowski, 1985] B. Moszkowski. A Temporal Logic for Multilevel Reasoning about Hardware. *IEEE Computer*, 18(2), 1985.

[Moszkowski, 1986] B. Moszkowski. *Executing Temporal Logic Programs*. Cambridge University Press, Cambridge, 1986.

[Mota *et al.*, 1997] E. Mota, D. Robertson, and A. Smaill. NatureTime: Temporal Granularity in Simulation of Ecosystems. *Journal of Symbolic Computation*, 22(5):665–698, 1997.

[Mourelatos, 1978] A. P. D. Mourelatos. Events, Processes and States. *Linguistics and Philosophy*, 2:415–434, 1978.

[Muller *et al.*, 1988] D. Muller, A. Saoudi, and P. Schupp. Weak Alternating Automata give a Simple Explanation of why most Temporal and Dynamic Logics are Decidable in Exponential Time. In *Proceedings of Third IEEE Symposium on Logic in Computer Science (LICS)*, pages 422–427. IEEE, 1988.

[Muller *et al.*, 1995] J.-C. Muller, J.-P. Lagrange, R. Weibel, and F. Salgé. Generalization: State of the Art and Issues. In J.-C. Muller, J.-P. Lagrange, and R. Weibel, editors, *GIS and Generalization*, pages 3–17. Taylor and Francis, London (GB), 1995.

[Muller, 1963] D. Muller. Infinite Sequences and Finite Machines. In *Proceedings of the Fourth Annual IEEE Symposium on Switching Circuit Theory and Logical Design*, pages 3–16, 1963.

[Murray, 1982] N. V. Murray. Completely Non-Clausal Theorem Proving. *Artificial Intelligence*, 18:67–85, 1982.

[Muscettola *et al.*, 1998] N. Muscettola, P. Nayak, B. Pell, and B. Williams. Remote Agent: To Boldly Go Where No AI System Has Gone Before. *Artificial Intelligence*, 103(1-2):5–48, 1998.

[Muscettola, 1994] N. Muscettola. HSTS: Integrating Planning and Scheduling. In M. Zweben and M.S. Fox, editors, *Intelligent Scheduling*, pages 169–212. Morgan Kaufmann, San Mateo, CA, 1994.

[Musen *et al.*, 1992] M.A. Musen, C.W. Carlson, L.M. Fagan, S.C. Deresinski, and E. H. Shortliffe. T–HELPER: Automated Support for Community–Based Clinical Research. In M.E. Frisse, editor, *Proceedings of the Sixteenth Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pages 719–723. McGraw Hill, 1992.

[Musen *et al.*, 1996] M.A. Musen, S.W. Tu, A.K. Das, and Y. Shahar. EON: A Component-Based Approach to Automation of Protocol-Directed Therapy. *Journal of the American Medical Informatics Association*, 3(6):367–388, 1996.

[Mylopoulos *et al.*, 1990] J. Mylopoulos, A. Borgida, M. Jarke, and M. Koubarakis. Telos: A Language for Representing Knowledge About Information Systems. *ACM Transactions on Information Systems*, 8(4):325–362, October 1990.

[Nau *et al.*, 1999] D. Nau, Y. Cao, A. Lotem, and H. Muñoz-Avila. SHOP: Simple Hierarchical Ordered Planner. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1999.

[Nau *et al.*, 2003] D. Nau, T.C. Au, O. Ilghami, U. Kuter, J.W. Murdoch, D. Woo, and F. Yaman. SHOP2: an HTN Planning Environment. *Journal of AI Research*, 2003.

[Navarrete and Marin, 1997a] I. Navarrete and R. Marin. Qualitative Temporal Reasoning with Points and Durations. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1454–1459. Morgan Kaufmann, 1997.

[Navarrete and Marin, 1997b] Isabel Navarrete and Roque Marin. Qualitative temporal reasoning with points and durations. In Martha E. Pollack, editor, *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Nagoya, Japan, August 1997. Morgan Kaufmann.

[Nebel and Bürckert, 1995] B. Nebel and H-J. Bürckert. Reasoning about Temporal Relations: A Maximal Tractable Subclass of Allen's Interval Algebra. *Journal of the ACM*, 42(1):43–66, 1995.

[Nebel, 1997] B. Nebel. Solving Hard Qualitative Temporal Reasoning Problems: Evaluating the Efficiency of Using the ORD-Horn Class. *CONSTRAINTS*, 1(3):175–190, 1997.

[Nebel, 2000] B. Nebel. On the Compilability and Expressive Power of Propositional Planning Formalisms. *Journal of Artificial Intelligence Research*, 12:271–315, 2000.

[Newton-Smith, 1980] W.H. Newton-Smith. *The Structure of Time*. Routledge & Heagan Paul, 1980.

[Newton, 1936] I. Newton. *Mathematical Principles of Natural Philosophy*. F. Cajori Ed., 1936.

[Ngo *et al.*, 1995] L. Ngo, P. Haddawy, and J. Helwig. A Theoretical Framework for Context-Sensitive Temporal Probability Model Construction with Application to Plan Projection. In *Proceedings on International Conference on Uncertainty in AI (UAI)*, 1995.

[Nguyen and Kambhampati, 2001] X. Nguyen and S. Kambhampati. Reviving Partial Order Planning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 459–466, 2001.

[Nguyen *et al.*, 1999] J. Nguyen, Y. Shahar, S.W. Tu, A.K. Das, and M.A. Musen. Integration of Temporal Reasoning and Temporal-Data Maintenance into a Reusable Database Mediator to Answer Abstract, Time-Oriented Queries: The Tzolkin System. *Journal of Intelligent Information Systems*, 13(1/2):121–145, 1999.

[Nicholson and Brady, 1994] A.E. Nicholson and J.M. Brady. Dynamic Belief Networks for Discrete Monitoring. *IEEE Transactions on Systems, Man and Cybernetics*, 24(11):1593–610, November 1994.

[Niemelä and Simons, 1997] I. Niemelä and P. Simons. Smodels – An Implementation of the Stable Model and Well-Founded Semantics for Normal Logic Programs. In J. Dix, U. Furbach, and A. Nerode, editors, *Proceedings of the Fourth International Conference on Logic Programming and Non-Monotonic Reasoning (LPNMR)*, pages 420–429. Springer, 1997.

[Niemelä, 1999] I. Niemelä. Logic Programs with Stable Model Semantics as a Constraint Programming Paradigm. *Annals of Mathematics and Artificial Intelligence*, 25(3-4):241–271, 1999.

[Niézette and Stevenne, 1992] M. Niézette and J.-M. Stevenne. An Efficient Symbolic Representation of Periodic Time. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM)*, pages 161–168, Baltimore, USA, 1992. ACM Press.

[Nishida and Doshita, 1987] T. Nishida and S. Doshita. Reasoning about Discontinuous Change. In *Proceedings of the Sixth (US) National Conference on Artificial Intelligence (AAAI)*, pages 643–648. AAAI press, 1987.

[Nitta *et al.*, 1988] K. Nitta, J. Nagao, and M. Tetsuya. A Knowledge Representation and Inference System for Procedural Law. *New Generation Computing*, 5:319–359, 1988.

[Nivat and Perrin, 1986] M. Nivat and D. Perrin. Ensembles Reconnaissables de mot Biinfinis. *Canadian Journal of Mathematics*, 38:513–537, 1986.

[Nökel, 1991] K. Nökel. *Temporally Distributed Symptoms in Technical Diagnosis*, volume 517. Springer-Verlag, Berlin, Heidelberg, New York, 1991.

[O'Connor *et al.*, 2002] M. J. O'Connor, S. W. Tu, and M. A. Musen. The Chronus II Temporal Database Mediator. In *Proceedings of the American Medical Informatics Association (AMIA) Annual Fall Symposium*, San Antonio, USA, 2002.

[Ohlbach, 1993] H-J. Ohlbach. Translation Methods for Non-Classical Logics – An Overview. *Journal of the IGPL*, 1(1):69–90, 1993.

[Ohno-Machado *et al.*, 1998] L. Ohno-Machado, J. Gennari, S. Murphy, and et al. The Guideline Interchange Format: A Model for Representing Guidelines. *Journal of the American Medical Informatics Association*, 5:357–72, 1998.

[Ohrstrom and Hasle, 1995] P. Ohrstrom and Per F. V. Hasle. *Temporal Logic: From Ancient Ideas to Artificial Intelligence*, volume 57 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

[Onder and Pollack, 1999] N. Onder and M. Pollack. Conditional Probabilistic Planning: A Unifying Algorithm and Effective Search Control Mechanisms. In *Proceedings of Sixtenth (US) National Conference on Artificial Intelligence (AAAI)*, 1999.

[Orgun, 1996] M.A. Orgun. On Temporal Deductive Databases. *Computational Intelligence*, 12(2):235–259, 1996.

[Özsoyŏglu and Snodgrass, 1995] R. Özsoyŏglu and T. Snodgrass. Temporal and Real-Time Databases: A Survey. *IEEE Transactions of Knowledge and Data Engineering*, 7:513–532, 1995.

[Paech, 1988] B. Paech. Gentzen-Systems for Propositional Temporal Logics. In E. Börger, H. Kleine Büning, and M. Richter, editors, *Proceedings of Conference on Computer Science Logic (CSL)*, volume 385 of *Lecture Notes in Computer Science*, pages 240–253, 1988.

[Papadias *et al.*, 1995] D. Papadias, Y. Theodoridis, T. Sellis, and M. Egenhofer. Topological Relations in the World of Minimum Bounding Rectangles: A Study with R-trees. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 92–103, 1995.

[Partee, 1997] B. Partee. Montague Grammar. In J. van Benthem and A. ter Meulen, editors, *Handbook of Logic and Language*, pages 5–91. MIT/Elsevier, 1997.

[Pearce and Wagner, 1989] D. Pearce and G. Wagner. Reasoning with Negative Information 1 – Strong Negation in Logic Programming. Technical report, Gruppe fur Logic, Wissentheorie and Information, Freie Universitat Berlin, Berlin, Germany, 1989.

[Pearl, 1988] J. Pearl. *Probablistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Mateo, CA, 1988.

[Pednault, 1986a] E. P. D. Pednault. Formulating Multiagent, Dynamic-World Problems in the Classical Planning Framework. In M.P. Georgeff and A.L. Lansky, editors, *Proceedings of the Timberline Oregon Workshop on Reasoning about Actions and Plans*, 1986.

[Pednault, 1986b] E. P. D. Pednault. Towards a Mathematical Theory of Plan Syntesis. Technical Report Ph.D. Thesis, Department of Electrical Engineering, Stanford University, Palo Alto, USA, 1986.

[Pednault, 1989] E. P. D. Pednault. ADL: Exploring the Middle Ground between STRIPS and the Situation Calculus. In *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 324–332, 1989.

[Pe'er and Shamir, 1997] I. Pe'er and R. Shamir. Satisfiability Problems on Intervals and Unit Intervals. *Theoretical Computer Science*, 175:349–372, 1997.

[Peirce, 1955] C. S. Peirce. *Philosophical Writings of Peirce*. Dover Publications, New York, 1955.

[Peleg *et al.*, 2001] M. Peleg, A. Boxwala, E. Bernstam, S.W. Tu, R.A. Greenes, and E.H. Shortliffe. Sharable Representation of Clinical Guidelines in GLIF: Relationship to the Arden Syntax. *Journal of Biomedical Informatics*, 34:170–181, 2001.

[Penberthy and Weld, 1992] J. Penberthy and D.S. Weld. UCPOP: a Sound, Complete, Partial-Order Planner for ADL. In *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 103–114, Los Altos, CA, 1992. Kaufmann.

[Penberthy and Weld, 1994] J. Penberthy and D. Weld. Temporal Planning with Continuous Change. In *Proceedings of Twelfth (US) National Conference on Artificial Intelligence (AAAI)*. AAAI/MIT Press, 1994.

[Penberthy, 1993] J. Penberthy. Planning with Continuous Change. Technical Report 93-12-01, Department of Computer Science & Engineering, University of Washington, USA, 1993.

[Peng and Reggia, 1990] Y. Peng and J.A. Reggia. *Abductive Inference Models for Diagnostic Problem Solving*. Springer–Verlag, 1990.

[Pereira *et al.*, 1990] L. Pereira, L. Caires, and J. Alferes. Classical negation in logic programs. In *7 Simposio Brasiliero de Inteligencia Artificial*, 1990.

[Perrin and Schupp, 1986] D. Perrin and P. E. Schupp. Automata on the Integers, Recurrence Distinguishability, and the Equivalence and Decidability of Monadic Theories. In *Proceedings of International Symposium on Logic in Computer Science (LICS)*, pages 301–304, Cambridge, USA, 16–18 June 1986. IEEE Computer Society.

[Perrin, 1990] D. Perrin. Finite Automata. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B. Elsevier, Amsterdam, 1990.

[Pinto and R.Reiter, 1993] J. Pinto and R.Reiter. Temporal Reasoning in Logic Programming: A Case for the Situation Calculus. In *Proceedings of the International Conference on Logic Programming (ICLP)*, pages 203–221, 1993.

[Pinto, 1994] J. Pinto. *Temporal Reasoning in the Situation Calculus*. PhD thesis, University of Toronto, Toronto, Ontario, Canada, February 1994.

[Pliuškevičius, 2001] R. Pliuškevičius. Deduction-Based Decision Procedure for a Clausal Miniscoped Fragment of FTL. In *Proceedings of the First International Joint Conference on Automated Reasoning (IJCAR)*, volume 2083 of *Lecture Notes in Artificial Intelligence*, pages 107–120. Springer, 2001.

[Pnueli, 1977] A. Pnueli. The Temporal Logic of Programs. In *Proceedings of the Eighteenth IEEE Symposium on the Foundations of Computer Science (FOCS)*, pages 46–57, Providence, USA, 31–2 1977. IEEE Computer Society Press.

[Pnueli, 1986] A. Pnueli. Specification and Development of Reactive Systems. In *Proceeings of Information Processing Conference*. Elsevier, 1986.

[Poesio and Brachman, 1991] M. Poesio and R.J. Brachman. Metric Constraints for Maintaining Appointments: Dates and Repeated Activities. In *Proceedings of the Nineth National Conference of the American Association for Artificial Intelligence*, pages 253–259. The MIT press, 1991.

[Poole *et al.*, 1998] D. Poole, A. Mackworth, and R. Goebel. *Computational Intelligence*. Oxford University Press, 1998.

[Poole, 1988] D. Poole. A Logical Framework for Default Reasoning. *Artifical Intelligence*, 36:27–47, 1988.

[Popkorn, 1994] S. Popkorn. *First Steps in Modal Logic*. Cambridge, 1994.

[Pople, 1973] H. Pople. On the Mechanization of Abductive Logic. In *Proceedings of the Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 147–152, 1973.

[Poulin *et al.*, 1992] D. Poulin, E. Mackaay, P. Bratley, and J. Fremont. Time Server — A Legal Time Specialist. In A. Martino, editor, *Expert Systems in Law*, pages 295–312, 1992.

[Prior, 1957] A. Prior. *Time and Modality*. Oxford University Press, 1957.

[Prior, 1967] A. Prior. *Past, Present and Future*. Oxford (Clarendon Press), 1967.

[Provan, 1993] G. Provan. Tradeoffs in Constructing and Evaluating Temporal Influence Diagrams. In *Proceedings of International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 40–47, 1993.

[Provetti, 1996] A. Provetti. Hypothetical Reasoning: From Situation Calculus to Event Calculus. *Computational Intelligence*, 12(3), 1996.

[Przymusinski, 1990] T. Przymusinski. Extended Stable Semantics for Normal and Disjunctive Programs. In D. Warren and P. Szeredi, editors, *Proceedings of the Seventh International Conference on Logic Programming (ICLP)*, pages 459–477, 1990.

[Psillos, 1996] S. Psillos. Ampliative Reasoning: Induction or Abduction. In *ECAI-96 workshop on Abductive and Inductive Reasoning*, 1996.

[Pujari and Sattar, 1999] K. Pujari and A. Sattar. A New Framework for Reasoning about Points, Intervals and Durations. In Thomas Dean, editor, *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1259–1267, Stockholm, Sweden, July 1999. Morgan Kaufmann.

[Pujari *et al.*, 1999] K. Pujari, G.V. Kumari, and A. Sattar. INDU: An Interval and Duration Network. In *Proceedings of Australian Joint Conference on Artificial Intelligence*, pages 291–303, 1999.

[Puppo and Dettori, 1995] E. Puppo and G. Dettori. Towards a Formal Model for Multi-Resolution Spatial Maps. *Lecture Notes in Computer Science*, 951:152–169, 1995.

[Rabideau *et al.*, 1999] G. Rabideau, R. Knight, S. Chien, A. Fukunaga, and A. Govindjee. Iterative Repair Planning for Spacecraft Operations in the ASPEN System. In *Proceedings of the International Symposium on Artificial Intelligence Robotics and Automation in Space (ISAIRAS)*, 1999.

[Rabin and Scott, 1959] M. Rabin and D. Scott. Finite Automata and their Decision Problem. *IBM Journal of Research*, 3:115–124, 1959.

[Rabin, 1969] M. Rabin. Decidability of Second-Order Theories and Automata on Infinite Trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969.

[Rabin, 1972] M. Rabin. *Automata on Infinite Objects and Church's Problem*. American Mathematical Society, 1972.

[Randell *et al.*, 1992] D. Randell, Z. Cui, and A. Cohn. A Spatial Logic based on Regions and Connection. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 165–176, 1992.

[Rao and Georgeff, 1991] A. S. Rao and M. P. Georgeff. Modeling Agents within a BDI-Architecture. In R. Fikes and E. Sandewall, editors, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, Cambridge, Massachusetts, April 1991. Morgan Kaufmann.

[Rao and Georgeff, 1993] A. S. Rao and M. P. Georgeff. A Model-Theoretic Approach to the Verification of Situated Reasoning Systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 318–324, Chambéry, France, 1993.

[Rao and Georgeff, 1995] A. S. Rao and M. Georgeff. BDI Agents: from Theory to Practice. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS)*, pages 312–319, San Francisco, USA, June 1995.

[Rautenberg, 1979] W. Rautenberg. *Klassische und Nichtklassische Aussagenlogik*. Vieweg, 1979.

[Rautenberg, 1983] W. Rautenberg. Modal Tableau Calculi and Interpolation. *Journal of Philosophical Logic*, 12:403–423, 1983.

[Rector, 2001] A. Rector. AIM: A Personal View of Where I Have Been and Where We Might Be Going. *Artificial Intelligence in Medicine*, 23:111–127, 2001.

[Reddy and Loveland, 1978] C. R. Reddy and D. W. Loveland. Presburger Arithmetic with Bounded Quantifier Alternation. In *Proceedings of the ACM Symposium on the Theory of Computing (STOC)*, pages 320–325, 1978.

[Reichgelt, 1987] H. Reichgelt. Semantics for Reified Temporal Logics. In J. Hallam and C. Mellish, editors, *Advances in Artificial Intelligence*, pages 49–61. John Wiley and Sons, 1987.

[Reichgelt, 1989] H. Reichgelt. A Comparison of First Order and Modal Logics of Time. In P. Jackson, H. Reichgelt, and F. van Harmelen, editors, *Logic-Based Knowledge Representation*, pages 143–176. MIT Press, 1989.

[Reiter, 1978] R. Reiter. On Closed World Data Bases. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 119–140. Plenum Press, New York, 1978.

[Reiter, 1980a] R. Reiter. A Logic for Default Reasoning. *Artificial Intelligence*, 13:81–132, 1980.

[Reiter, 1980b] R. Reiter. Equality and Domain Closure in First-Order Databases. *Journal of the ACM*, 27:235–249, 1980.

[Reiter, 1984] R. Reiter. Towards a Logical Reconstruction of Relational Database Theory. In M. Brodie, J. Mylopoulos, and J. Schmidt, editors, *On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages*, pages 191–233. Springer Verlag, 1984.

[Reiter, 1988] R. Reiter. On Integrity Constraints. In *Proceedings of the Second Conference on Theoretical Aspects of Reasoning About Knowledge (TARK)*, pages 97–111, Asilomar, USA, 1988.

[Reiter, 1991] R. Reiter. The Frame Problem in the Situation Calculus: A simple Solution (Sometimes) and a Completeness Result for Goal Regression. In V. Lifschitz, editor, *Artificial Intelligence and Mathematical Theory of Computation: Papers in Honour of John McCarthy*, pages 359–380. Academic Press, 1991.

[Reiter, 2001] R. Reiter. *Knowledge in Action*. MIT press, 2001.

[Renz and Nebel, 1997] J. Renz and B. Nebel. On the Complexity of Qualitative Spatial Reasoning: A Maximal Tractable Fragment of the Region Connection Calculus. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 522–527. Morgan Kaufmann, 1997.

[Rescher and Garson, 1968] N. Rescher and J. Garson. Topological Logic. *Journal of Symbolic Logic*, 33:537–548, 1968.

[Rescher and Urquhart, 1971] N. Rescher and A. Urquhart. *Temporal Logic*. Library of Exact Philosophy. Springer Verlag, 1971.

[Reynolds, 1992] M. Reynolds. An Axiomatization for Until and Since over the Reals without the IRR rule. *Studia Logica*, 51:165–193, May 1992.

[Reynolds, 1994] M. Reynolds. Axiomatizing U and S over Integer Time. In D. Gabbay and H.-J. Ohlbach, editors, *Proceedings of the First International Conference on Temporal Logic (ICTL)*, volume 827 of *Lecture Notes in Artificial Intelligence*, pages 117–132, Bonn, Germany, July 1994. Springer-Verlag.

[Reynolds, 1996] M. Reynolds. Axiomatising First-Order Temporal Logic: Until and Since over Linear Time. *Studia Logica*, 57:279–302, 1996.

[Reynolds, 1998] M. Reynolds. A Decidable Logic of Parallelism. *Notre Dame Journal of Formal Logic*, 1998.

[Reynolds, 2000] M. Reynolds. More Past Glories. In *Proceedings of Fifteenth Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 229–240. IEEE, June 2000.

[Reynolds, 2001] M. Reynolds. An Axiomatization of Full Computation Tree Logic. *Journal of Symbolic Logic*, 66(3):1011–1057, 2001.

[Reynolds, 2003] M. Reynolds. An Axiomatization of PCTL*. *Information and Computation*, 2003.

[Rigaux and Scholl, 1995] P. Rigaux and M. Scholl. Multi-Scale Partitions: Application to Spatial and Statistical Databases. In *Proceedings of the Fourth International Symposium on Spatial Databases (SSD)*, volume 951 of *Lecture Notes in Computer Science*, pages 170–183. Springer-Verlag, 1995.

[Rinner and Kuipers, 1999] B. Rinner and B. Kuipers. Monitoring Piecewise Continuous Behaviors by Refining Semi-Quantative Trackers. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1080–1086, 1999.

[Rintannen, 1999] J. Rintannen. Constructing Conditional Plans by a Theorem Prover. *Journal of AI Research*, 10:323–352, 1999.

[Rit, 1986] J. F. Rit. Propagating Temporal Constraints for Scheduling. In *Proceedings of the Fifth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 383–388. Morgan Kaufmann, 1986.

[Robinson, 1965] J. A. Robinson. A Machine–Oriented Logic Based on the Resolution Principle. *Journal of the ACM*, 12(1):23–41, January 1965.

[Roman, 1990] G. Roman. Formal Specification of Geographic Data Processing Requirements. *IEEE Transaction on Knowledge and Data Engineering*, 2(4):177–192, 1990.

[Rosner and Pnueli, 1986] R. Rosner and A. Pnueli. A Choppy Logic. In *Proceedings of the Symposium on Logic in Computer Science (LICS)*, pages 306–313. IEEE Computer Society, June 1986.

[Russ, 1989] T. A. Russ. Using Hindsight in Medical Decision Making. In *Proceedings of the Symposium on Computer Applications in Medical Care (SMAMC)*, pages 38–44, New York, USA, 1989. IEEE Computer Society Press.

[Russ, 1995] T. A. Russ. Use of Data Abstraction Methods to Simplify Monitoring. *Artificial Intelligence in Medicine*, 7:497–514, 1995.

[Russell and Norvig, 1995] S. Russell and P. Norvig. *Artificial Intelligence: a Modern Approach*. Prentice Hall, 1995.

[Russell, 1956] B. Russell. On Order in Time. In R.C. Marsh, editor, *Bertrand Russell: Logic and Knowledge (essays 1901–1950)*. Routledge, 1956.

[Sacerdoti, 1975] E. D. Sacerdoti. The Nonlinear Nature of Plans. In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence (IJCA)*, pages 206–214, Tbilisi, Georgia, USSR, September 1975.

[Sadri and Kowalski, 1995] F. Sadri and R. Kowalski. Variants of the Event Calculus. In L. Sterling, editor, *Proceedings of the International Conference on Logic Programming (ICLP)*, 1995.

[Safra, 1988] S. Safra. On the Complexity of ω-Automata. In *Proceedings of Twenty-Ninth IEEE Symposium on the Foundations of Computer Science (FOCS)*, 1988.

[Salzberg and Tsotras, 1999] B. Salzberg and V. J. Tsotras. Comparison of Access Methods for Time-Evolving Data. *ACM Computing Surveys*, 31(2):158–221, 1999.

[Sandewall, 1994] E. Sandewall. *Features and Fluents: The Representation of Knowledge about Dynamical Systems*, volume I. Oxford University Press, 1994.

[Schild, 1991] K. Schild. A Correspondence Theory for Terminological Logics: Preliminary Report. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 466–471, Sidney, Australia, 1991.

[Schild, 1993] K. Schild. Combining Terminological Logics with Tense Logic. In M. Filgueiras and L. Damas, editors, *Progress in Artificial Intelligence – Proceedings of the Sixth Portuguese Conference on Artificial Intelligence (EPIA)*, volume 727 of *Lecture Notes in Computer Science*, Porto, Portugal, October 1993. Springer.

[Schmiedel, 1990] A. Schmiedel. A Temporal Terminological Logic. In *Proceedings of the Eighth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 640–645. AAAI Press/The MIT Press, 1990.

[Schrag and Boddy, 1991] B. Schrag and M. Boddy. β-TMM Functional Description. Technical report, Honeywell SRC, USA, 1991.

[Schrag *et al.*, 1992] R. Schrag, M. Boddy, and J. Carciofini. Managing Disjunction for Practical Temporal Reasoning. In Swartout and Nebel [1992], pages 36–46.

[Schrijver, 1986] A. Schrijver, editor. *Theory of Integer and Linear Programming*. Wiley, 1986.

[Schubert and Hwang, 1989] L.K. Schubert and C. H. Hwang. An Episodic Knowledge Representation for Narrative Texts. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 444–458, Toronto, Canada, 1989. Morgan Kaufmann.

[Schubert *et al.*, 1987] L. K. Schubert, M. A. Papalaskaris, and J. Taugher. Accelerating Deductive Inference: Special Methods for Taxonomies, Colours, and Times. In N. Cercone and G. McCalla, editors, *The Knowledge Frontier: Essays in the Representation of Knowledge*, pages 187–220. Springer-Verlag, Berlin, Heidelberg, New York, 1987.

[Schubert, 1990] L. Schubert. Monotonic Solution of the Frame Problem in the Situation Calculus: An Efficient Method for Worlds with Fully Specified Actions. In *Knowledge Representation and Defeasible Reasoning*, pages 23–67. Kluwer Academic Press, 1990.

[Schwalb and Dechter, 1997] E. Schwalb and R. Dechter. Processing Disjunctions in Temporal Constraint Networks. *Artificial Intelligence*, 93:29–61, 1997.

[Schwalb and Vila, 1998] E. Schwalb and L. Vila. Processing Metric Temporal Constraints, 1998.

[Schwalb *et al.*, 1994] E. Schwalb, K. Kask, and R. Dechter. Temporal Reasoning with Constraints on Fluents and Events. In *Proceedings of (US) National Conference on Artificial Intelligence (AAAI)*. AAAI press, 1994.

[Schwalb *et al.*, 1996] E. Schwalb, L. Vila, and R. Dechter. Temporal Constraint Logic Programming. Technical report, UC Irvine, California, USA, 1996.

[Schwalb, 1996] E. Schwalb. Personal Communication, 1996.

[Schwendimann, 1998a] S. Schwendimann. A New One-Pass Tableau Calculus for PLTL. In H. de Swart, editor, *Proceedings of Tableaux Workshop*, volume 1397 of *Lecture Notes in Artificial Intelligence*, pages 277–291. Springer-Verlag, 1998.

[Schwendimann, 1998b] S. Schwendimann. *Aspects of Computational Logic*. PhD thesis, University of Bern, Switzerland, 1998.

[Scott, 2002] S. L. Scott. Bayesian Methods for Hidden Markov Models. Recursive Computing in the 21st Century. *Journal of the American Statistical Assoication*, 97:337–351, 2002.

[Sergot, 1988] M. Sergot. Representing Legislation as Logic Programs. In Hayes, Michie, and Richards, editors, *Machine Intelligence*, pages 209–260. Oxford University Press, 1988.

[Sergot, 1995] M. Sergot. Using Logic for Knowledge Representation in Legal Knowledge Based System. Tutorial Notes from the Fifth International Conference on Artificial Intelligence in Law, May 1995.

[Shahar and Cheng, 1999] Y. Shahar and C. Cheng. Intelligent Visualization and Exploration of Time-Oriented Clinical Data. *Topics in Health Information Systems*, 20(2):15–31, 1999.

[Shahar and Cheng, 2000] Y. Shahar and C. Cheng. Model-Based Visualization of Temporal Abstractions. *Computational Intelligence*, 16(2):279–306, 2000.

[Shahar and Molina, 1998] Y. Shahar and M. Molina. Knowledge-Based Spatiotemporal Linear Abstraction. *Pattern Analysis and Applications*, 1(2):91–104, 1998.

[Shahar and Musen, 1996] Y. Shahar and M.A. Musen. Knowledge-Based Temporal Abstraction in Clinical Domains. *Artificial Intelligence in Medicine*, 8(3):267–298, 1996.

[Shahar *et al.*, 1998] Y. Shahar, S. Miksch, and P.D. Johnson. The Asgaard Project: A Task-Specific Framework for the Application and Critiquing of Time-Oriented Clinical Guidelines. *Artificial Intelligence in Medicine*, 14:29–51, 1998.

[Shahar *et al.*, 1999] Y. Shahar, H. Chen, D. Stites, L. Basso, H. Kaizer, D. Wilson, and M.A. Musen. Semiautomated Acquisition of Clinical Temporal-Abstraction Knowledge. *Journal of the American Medical Informatics Association*, 6(6):494–511, 1999.

[Shahar *et al.*, 2003a] Y. Shahar, D. Boaz, G. Tahan, M. Galperin, D. Goren-Bar, H. Kaizer, L.V. Basso, S.B. Martins, and M.K. Goldstein. Interactive Visualization and Exploration of Time-Oriented Clinical Data using a Distributed Temporal-Abstraction Architecture. In *Proceedings of the AMIA Annual Fall Symposium*, Washington (DC), USA., 2003.

[Shahar *et al.*, 2003b] Y. Shahar, E. Shalom, A. Mayaffit, O. Young, M. Galperin, S.B. Martins, and M.K. Goldstein. A Distributed, Collaborative, Structuring Model for a Clinical-Guideline Digital-Library. In *Proceedings of the AMIA Annual Fall Symposium*, Washington (DC), USA., 2003.

[Shahar *et al.*, 2003c] Y. Shahar, O. Young, E. Shalom, A. Mayaffit, R. Moskovitch, A. Hessing, and M. Galperin. DEGEL: A Hybrid, Multiple-Ontology Framework for Specification and Retrieval of Clinical Guidelines. In *Proceedings of the Ninth Conference on Artificial Intelligence in Medicine — Europe (AIME)*, Protaras, Cyprus, 2003.

[Shahar, 1997] Y. Shahar. A Framework for Knowledge-Based Temporal Abstraction. *Artificial Intelligence*, 90(1):79–133, 1997.

[Shahar, 1998] Y. Shahar. Dynamic Temporal Interpretation Contexts for Temporal Abstraction. *Annals of Mathematics and Artificial Intelligence*, 22(1–2):159–192, 1998.

[Shahar, 1999] Y. Shahar. Knowledge-Based Temporal Interpolation. *Journal of Experimental and Theoretical Artificial Intelligence*, 11:123–144, 1999.

[Shanahan and Southwick, 1989] M. Shanahan and R. Southwick. *Search, Inference and Dependencies in Artificial Intelligence*. Ellis Horwood, New York, Chichester,Brisbane, Toronto, 1989.

[Shanahan, 1987] M. Shanahan. *Solving the Frame Problem: A Mathematical Investigation of the Common Sense Law of Inertia*. MIT Press, 1987.

[Shanahan, 1989] M. Shanahan. Prediction is Deduction but Explanation is Abduction. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, page 1055, 1989.

[Shanahan, 1990] M. Shanahan. Representing Continuous Change in the Event Calculus. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, page 598, 1990.

[Shanahan, 1999] M. P. Shanahan. The Event Calculus Explained. In M. J. Wooldridge and M. Veloso, editors, *Artificial Intelligence Today*, volume 1600 of *Lecture Notes in Artificial Intelligence*, pages 409–430. Springer, 1999.

[Sherman *et al.*, 1995] E.H. Sherman, G. Hripcsak, J. Starren, R.A. Jender, and P. Clayton. Using Intermediate States to Improve the Ability of the Arden Syntax to Implement Care Plans and Reuse Knowledge. In R. M. Gardner, editor, *Proceedings of the Annual Symposium on Computer Applications in Medical Care (SCAMC)*, pages 238–242, New Orleans, USA, 1995. Hanley & Belfus.

[Shiffman *et al.*, 2000] R. Shiffman, B. Karras, A. Agrawal, R. Chen, L. Marenco, and S. Nath. GEM: A Proposal for a More Comprehensive Guideline Document Model using XML. *Journal of the American Medical Informatics Association*, 7(5):488–498, 2000.

[Shoham and McDermott, 1988] Y. Shoham and D. McDermott. Problems in Formal Temporal Reasoning. *Artificial Intelligence*, 36(1):49–61, August 1988.

[Shoham, 1987] Y. Shoham. Temporal Logics in AI: Semantical and Ontological Considerations. *Artificial Intelligence*, 33:89–104, 1987.

[Shoham, 1988] Y. Shoham. *Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence*. MIT Press, 1988.

[Shoham, 1993] Y. Shoham. Agent-Oriented Programming. *Artificial Intelligence*, 60(1):51–92, 1993.

[Shostak, 1981] R. Shostak. Deciding Linear Inequalities by Computing Loop Residues. *Journal of the ACM*, 28(4):769–779, 1981.

[Shults and Kuipers, 1997] B. Shults and B. Kuipers. Proving Properties of Continuous Systems: Qualitative Simulation and Temporal Logic. *Artificial Intelligence*, 92:91–129, 1997.

[Singh, 1994] M. P. Singh. *Multiagent Systems: A Theoretical Framework for Intentions, Know-How, and Communications*, volume 799 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1994.

[Sistla and Clarke, 1985] A. Sistla and E. Clarke. Complexity of Propositional Linear Temporal Logics. *Journal of the ACM*, 32:733–749, 1985.

[Sistla *et al.*, 1987] A. Sistla, M. Vardi, and P. Wolper. The Complementation Problem for Buchi Automata with Applications to Temporal Logic. *Theoretical Computer Science*, 49:217–237, 1987.

[Sistla *et al.*, 1997] A. P. Sistla, O. Wolfson, S. Chamberlain, and S. Dao. Modeling and Querying Moving Objects. In *Proceedings of IEEE International Conference on Data Engineering*, pages 422–432, 1997.

[Skiadopoulos, 2002] Spiros Skiadopoulos. *Query Evaluation in Spatial Constraint Databases*. PhD thesis, Dept. of Electrical and Computer Engineering, National Technical University of Athens, 2002.

[Skyt *et al.*, 2003] J. Skyt, C. S. Jensen, and L. Mark. A Foundation for Vacuuming Temporal Databases . *Data and Knowledge Engineering*, 44(1):1–29, 2003.

[Smith and Weld, 1999] D. E. Smith and D. S. Weld. Temporal Planning with Mutual Exclusion Reasoning. In *Proceedings of the Internationional Joint Conference on Artificial Intelligence (IJCAI)*, pages 326–337, 1999.

[Smith, 1980] R. G. Smith. *A Framework for Distributed Problem Solving*. UMI Research Press, 1980.

[Smullyan, 1968] R. Smullyan. *First-order Logic*. Springer, 1968.

[Smyth *et al.*, 1997] P. Smyth, D. Heckerman, and M.I. Jordan. Probabilistic Independence Networks for Hidden Markov Models. *Neural Computation*, 9:227–269, 1997.

[Snodgrass and Ahn, 1986] R. Snodgrass and I. Ahn. Temporal Databases. *IEEE Computer*, 19(9):35–42, 1986.

[Snodgrass *et al.*, 1994] R. T. Snodgrass, I. Ahn, G. Ariav, D. Batory, J. Clifford, C. E. Dyreson, R. Elmasri, F. Grandi, C. S. Jensen, W. Kafer, N. Kline, K. Kulkarni, T. Y. C. Leung, N. Lorentzos, J. F. Roddick, A. Segev, M. D. Soo, and S. A. Sripada. TSQL2 Language Specification. *SIGMOD Record*, 23(1):65–86, March 1994.

[Snodgrass *et al.*, 1995] R. T. Snodgrass, C. S. Jensen, and M. H. Böhlen. Evaluating and Enhancing the Completeness of TSQL2. Technical Report TR 95-5, Computer Science Department, University of Arizona, USA, 1995.

[Snodgrass *et al.*, 1996] R. T. Snodgrass, M. H. Böhlen, C. S. Jensen, and A. Steiner. Adding Valid Time to SQL/Temporal. ISO/IEC JTC1/SC21/WG3 DBL MAD-146r2 21/11/96, (change proposal), International Organization for Standardization, 1996.

[Snodgrass, 1987] R. T. Snodgrass. The Temporal Query Language TQuel. *ACM Transactions on Database Systems*, 12(2):247–298, June 1987.

[Snodgrass, 1990] R. T. Snodgrass. Temporal Databases: Status and Research Direction. *SIGMOD RECORD*, 19:83–89, 1990.

[Snodgrass, 1993] R. T. Snodgrass. An Overview of TQuel. In *Temporal Databases: Theory, Design, and Implementation*, chapter 6, pages 141–182. Benjamin/Cummings, 1993.

[Snodgrass, 1995] R. T. Snodgrass, editor. *The TSQL2 Temporal Query Language*. Kluwer Academic Publishers, 1995.

[Snodgrass, 1999] R. T. Snodgrass. *Developing Time-Oriented Database Applications in SQL*. Morgan Kaufmann, 1999.

[Son *et al.*, 2001] T. Son, C. Baral, and S. McIlraith. Extending Answer Set Planning with Sequence, Conditionals, Loops, Non-Deterministic Choice and Procedural Constructs. In *Proceedings of the AAAI Spring symposium on Answer Set Programming*, 2001.

[Song and Cohen, 1988] F. Song and R. Cohen. The Interpretation of Temporal Relations in a Narrative. In *Proceedings of the Seventh National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 745–750. Morgan Kaufmann, 1988.

[Song and Cohen, 1991] F. Song and R. Cohen. Tense Interpretation in the Context of Narrative. In *Proceedings of the Nineth National Conference of the American Association for Artificial Intelligence*, pages 131–136. MIT Press, 12–19 July 1991.

[Song and Cohen, 1996] F. Song and R. Cohen. A Strengthened Algorithm for Temporal Reasoning about Plans. *Computational Intelligence*, 12(2):331–356, 1996.

[Sontag, 1985] E. Sontag. Real Addition and the Polynomial Time Hierarchy. *Information Processing Letters*, 20:115–120, 1985.

[Sripada *et al.*, 1994] S. Sripada, B. Rosser, J. Bedford, and R. Kowalski. Temporal Database Technology for Air Traffic Flow Management. In *Proceedings of First International Conference on Applications of Databases (ADB)*, volume 819 of *Lecture Notes in Computer Science*. Springer-Verlag, 1994.

[Staab, 1998] S. Staab. On Non-Binary Temporal Relations. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 567–571, 1998.

[Stillman *et al.*, 1993] J. Stillman, R. Arthur, and A. Deitsch. Tachyon: A Constraint-Based Temporal Reasoning Model and its Implementation. *SIGART Bulletin*, 4(3), 1993.

[Stirling, 1992] C. Stirling. Modal and Temporal Logics. In D.M. Gabbay, S. Abramsky, and T.S.E Maibaum, editors, *Handbook of Logic in Computer Science*, volume 2, pages 477–563. Clarendon Press, Oxford, 1992.

[Stockmeyer, 1977] L.J. Stockmeyer. The Polynomial-Time Hierarchy. *Theoretical Computer Science*, 3:1–22, 1977.

[Street, 1982] R. Street. Propositional Dynamic Logic of Looping and Converse. *Information and Control*, 54:121–141, 1982.

[Streett and Emerson, 1984] R.S. Streett and E.A. Emerson. The Propositional mu-Calculus is Elementary. In J. Paredaens, editor, *Proceedings of the Eleventh International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 172, pages 465–472. Springer-Verlag, Antwerp, Belgium, July 1984.

[Stroetman, 1993] K. Stroetman. A Completeness Result for SLDNF-Resolution. *Journal of Logic Programming*, 15:337–355, 1993.

[Sturm and Wolter, 2000] H. Sturm and F. Wolter. A Tableau Calculus for Temporal Description Logic: The Expanding Domain Case. *Journal of Logic and Computation*, 2000.

[Subrahmanian and Zaniolo, 1995] V. Subrahmanian and C. Zaniolo. Relating Stable Models and AI Planning Domains. In L. Sterling, editor, *Proceedings of the International Conference on Logic Programming (ICLP)*, pages 233–247. MIT Press, 1995.

[Sussman, 1990] G. J. Sussman. The Virtuous Nature of Bugs. In J. Allen, J. Hendler, and A. Tate, editors, *Readings in Planning*, chapter 3, pages 111–117. Morgan Kaufmann Publishers, Inc., 1990.

[Swartout and Nebel, 1992] B. Swartout and B. Nebel, editors. *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR)*, Cambridge, MA, USA, October 1992. Morgan Kaufmann.

[Szalas, 1987] A. Szalas. A Complete Axiomatic Characterization of First-Order Temporal Logic of Linear Time. *Theoretical Computer Science*, 54:199–214, 1987.

[Tang and Young, 2000] P. Tang and C. Young. ActiveGuidelines: Integrating Web-Based Guidelines with Computer-Based Patient Records. In M.J Overhage, editor, *Proceedings of the AMIA Annual Symposium*, Los Angeles, USA, 2000. Hanley & Belfus.

[Tansel *et al.*, 1993] A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. T. Snodgrass, editors. *Temporal Databases: Theory, Design, and Implementation*. Benjamin/Cummings, 1993.

[Tansel, 1993] A. Tansel. A Generalized Relational Framework for Modelling Temporal Data. In Tansel et al. [1993], pages 183–201.

[Tarjan, 1972] R. Tarjan. Depth First Search and Linear Graph Algorithms. *SIAM Journal of Computing*, 1(2):215–225, 1972.

[Tarski, 1941] A. Tarski. On the Calculus of Relations. *Journal of Symbolic Logic*, 6:73–89, 1941.

[Tate, 1977] A. Tate. Generating project networks. In *Proceedings of IJCAI-77*, 1977.

[ter Meulen and Smessaert, 2004] A. ter Meulen and H. Smessaert. Dynamic Temporal Reasoning with Aspectual Adverbs. To appear in *Linguistics and Philosophy*, 2004.

[ter Meulen, 1990] A. ter Meulen. English Aspectual Verbs as Generalized Quantifiers. In J. Carter et al., editor, *NELS 20*, pages 378–390. GLSA, Department of Linguistics, University of Massachusetts, USA, 1990.

[ter Meulen, 1995] A. ter Meulen. *Representing Time in Natural Language – The dynamic interpretation of tense and aspect*. Bradford Books, MIT Press, Cambridge (Mass.), 1995.

[ter Meulen, 2000] A. ter Meulen. Chronoscopes: Dynamic Tools for Temporal Reasoning. In J. Higginbotham and F. Pianesi, editors, *Speaking of Events*, pages 151–168. Oxford University Press, Oxford, 2000.

[ter Meulen, 2003] A. ter Meulen. Situated Reasoning in Time about Time. In B. Löwe et al., editor, *Foundations of the Formal Sciences II, Applications of Mathematical Logic in Philosophy and Linguistics*, volume 17 of *Trends in Logic*. Kluwer Academic Press, Dordrecht, 2003.

[Terenziani, 1996] P. Terenziani. Toward an Ontology Dealing with Periodic Events. In W Wahlster, editor, *Proceedings of the Fifteenth European Conference on Artificial Intelligence (ECAI)*, pages 43–47. John Wiley & Sons, 1996.

[Thatcher and Wright, 1968] J. Thatcher and J. Wright. Generalized Finite Automata Theory with an Application to a Decision Problem of Second-Order Logic. *Mathematical Systems Theory*, 2(1):57–81, 1968.

[The STREAM Group, 2003] The STREAM Group. STREAM: The Stanford Stream Data Manager (short overview paper). *IEEE Data Engineering Bulletin*, 26(1), 2003.

[Thiébaux *et al.*, 1996] S. Thiébaux, M.-O. Cordier, O. Jehl, and J.-P. Krivine. Supply restoration in Power Distribution Systems – A Case Study in Integrating Model-Based Diagnosis and Repair Planning. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence (UAI)*. Morgan Kaufmann, 1996.

[Thielscher, 1997] M. Thielscher. Ramification and Causality. *Artificial Intelligence*, 89(1-2):317–364, 1997.

[Thomas, 1990] W. Thomas. Automata on Infinite Objects. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B. Elsevier, Amsterdam, 1990.

[Thornton *et al.*, 2002] J. Thornton, M. Beaumount, A. Sattar, and M. Maher. Applying Local Search to Temporal Reasoning. In *Proceedings of the Ninth International Symposium on Temporal Representation and Reasoning (TIME)*, Manchester, UK, 2002. IEEE Computer Society.

[Thorthon *et al.*, 2004] J. Thorthon, M. Beaumont, A. Sattar, and M. Maher. A Local Search Approach to Modelling and Solving Interval Algebra Problems. *Journal of Logic and Computation*, 14(1):93–112, 2004.

[Tobies, 2000] S. Tobies. The Complexity of Reasoning with Cardinality Restrictions and Nominals in Expressive Description Logics. *Journal of Artificial Intelligence Research*, 12:199–217, 2000.

[Toman and Chomicki, 1998] D. Toman and J. Chomicki. Datalog with Integer Periodicity Constraints. *Journal of Logic Programming*, 35(3):263–306, 1998.

[Toman and Niwinski, 1996] D. Toman and D. Niwinski. First-Order Queries over Temporal Databases Inexpressible in Temporal Logic. In *International Conference on Extending Database Technology (EDBT)*, Avignon, France, 1996.

[Toman, 1996] D. Toman. Point vs. Interval-based Query Languages for Temporal Databases. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, Montréal, Canada, June 1996.

[Toman, 1997] D. Toman. Point-based Temporal Extensions of SQL. In *International Conference on Deductive and Object-Oriented Databases*, 1997.

[Toman, 2001] D. Toman. Expiration of Historical Databases. In *Proceedings of International Symposium on Temporal Representation and Reasoning (TIME)*, pages 128–135. IEEE Press, 2001.

[Toman, 2003a] D. Toman. Logical Data Expiration. In Chomicki et al. [2003b], chapter 7, pages 203–238.

[Toman, 2003b] D. Toman. Logical Data Expiration for Fixpoint Extensions of Temporal Logics. In *International Symposium on Advances in Spatial and Temporal Databases (SSTD)*, pages 380–393, 2003.

[Toman, 2003c] D. Toman. On Incompleteness of Multi-dimensional First-order Temporal Logics. In *Proceedings of International Symposium on Temporal Representation and Reasoning and International Conference on Temporal Logic (TIME-ICTL)*, pages 99–106, 2003.

[Topaloglou, 1996] T. Topaloglou. *On the Representation of Partial Spatial Information in Knowledge Bases*. PhD thesis, University of Toronto, Toronto, USA, 1996.

[Trinquart and Ghallab, 2001] R. Trinquart and M. Ghallab. An Extended Functional Representation in Temporal Planning: Towards Continuous Change. In *Proceedings of European Conference on Planning (ECP)*, 2001.

[Tsamardinos and Pollack, 2003] I. Tsamardinos and M.E. Pollack. Efficient Solution Techniques for Disjunctive Temporal Reasoning Problems. *Artificial Intelligence*, 151(1-2):43–90, 2003.

[Tsang, 1986] E. P. K. Tsang. Plan Generation in a Temporal Frame. In *Proceedings of the Seventh European Conference on Artificial Intelligence (ECAI)*, pages 479–493, 1986.

[Tsang, 1987a] E. P. K. Tsang. Time Structures for AI. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 456–461. Morgan Kaufmann, 1987.

[Tsang, 1987b] E. P. K. Tsang. TLP- A Temporal Planner. In J. Hallam and C. Mellish, editors, *Advances in Artificial Intelligence*, pages 63–78. John Wiley&sons, 1987.

[Tu *et al.*, 1989] S.W. Tu, M.G. Kahn, M.A. Musen, J.C. Ferguson, E.H. Shortliffe, , and L. M. Fagan. Episodic Skeletal-Plan Refinement on Temporal Data. *Communications of the ACM*, 32:1439–1455, 1989.

[Tu *et al.*, 1995] S.W. Tu, H. Eriksson, J.H. Gennari, and M.A. Shahar, Y.and Musen. Ontology-Based Configuration of Problem-Solving Methods and Generation of Knowledge-Acquisition Tools: Application of PROTÉGÉ-II to Protocol-Based Decision Support. *Artificial Intelligence in Medicine*, 7(3):257–289, 1995.

[Turner, 1994] H. Turner. Signed Logic Programs. In *Proceedings of the International Symposium on Logic Programming (ILPS)*, pages 61–75, 1994.

[Turner, 1997] H. Turner. Representing Actions in Logic Programs and Default Theories. *Journal of Logic Programming*, 31(1-3):245–298, 1997.

[Tuzhilin and Clifford, 1990] A. Tuzhilin and J. Clifford. A Temporal Relational Algebra as a Basis for Temporal Relational Completeness. In *International Conference on Very Large Data Bases (VLDB)*, 1990.

[Ullman, 1989] J.D. Ullman. *Principles of Database and Knowledge-Base Systems*, volume 2. Computer Science Press, 1989.

[Valdes *et al.*, 1982] J. Valdes, R.E. Tarjan, and E.L. Lawler. The Recognition of Series Parallel Digraphs. *SIAM Journal of Computing*, 11(2):298–313, 1982.

[van Beek and Cohen, 1990] P. van Beek and R. Cohen. Exact and Approximate Reasoning about Temporal Relations. *Computational Intelligence*, 6(3):132–144, 1990.

[van Beek and Manchak, 1996] P. van Beek and D.W. Manchak. The Design and Experimental Analysis of Algorithms for Temporal Reasoning. *Journal of Artificial Intelligence Research*, 4:1–18, 1996.

[van Beek, 1990] P. van Beek. Reasoning about Qualitative Temporal Information. In *Proceedings of the Eighth National Conference of the American Association for Artificial Intelligence (AAAI)*, pages 728–734, Boston, MA, 1990.

[van Beek, 1991] P. van Beek. Temporal Query Processing with Indefinite Information. *Artificial Intelligence in Medicine*, 3:325–339, 1991.

[van Beek, 1992] P. van Beek. Reasoning about Qualitative Temporal Information. *Artificial Intelligence*, 58(1-3):297–321, 1992.

[Van Belleghem *et al.*, 1994] K. Van Belleghem, M. Denecker, and D. De Schreye. The Abductive Event Calculus as a General Framework for Temporal Databases. In *Proceedings of the First International Conference on Temporal Logic (ICTL)*, pages 301–316, 1994.

[Van Belleghem *et al.*, 1995] K. Van Belleghem, M. Denecker, and D. De Schreye. Combining Situation Calculus and Event Calculus. In *Proceedings of the Twelfth International Conference on Logic Programming (ICLP)*, pages 83–97, 1995.

[Van Bemmel, 1996] J.H. Van Bemmel. Medical Informatics, Art or Science? *Methods of Information in Medicine*, 35:157–117, 1996.

[van Benthem, 1983] J.F.A.K. van Benthem. *The Logic of time*. D. Reidel Publishing Company, Dordrecht, 1983.

[van Benthem, 1991] J.F.A.K. van Benthem. *The Logic of Time: A Model-Theoretic Investigation into the Varieties of Temporal Ontology and Temporal Discourse (second edition)*. (Synthese Library, Vol 156) Kluwer Academic Publishers, Dordrecht, 1991.

[van der Meyden, 1992] R. van der Meyden. The Complexity of Querying Indefinite Data About Linearly Ordered Domains (Preliminary Version). In *Proceedings of the Eleventh ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 331–345, 1992.

[van Gelder *et al.*, 1991] A. van Gelder, K. Ross, and J. Schlipf. The Well-Founded Semantics for General Logic Programs. *Journal of the ACM*, 38(3):620–650, 1991.

[van Linder *et al.*, 1996] B. van Linder, W. van der Hoek, and J. J. Ch. Meyer. How to Motivate Your Agents. In M. Wooldridge, J. P. Müller, and M. Tambe, editors, *Intelligent Agents II*, volume 1037 of *Lecture Notes in Artificial Intelligence*, pages 17–32. Springer-Verlag, 1996.

[Vardi and Stockmeyer, 1985] M. Vardi and L. Stockmeyer. Improved Upper and Lower Bounds for Modal Logics of Programs. In *Proceedings of the Seventeenth ACM Symposium on the Theory of Computing (STOC)*, pages 240–251. ACM Press, 1985.

[Vardi and Wolper, 1986] M.Y. Vardi and P. Wolper. An Automata-Theoretic Approach to Automatic Program Verification (Preliminary Report). In *Proceedings of the First IEEE Symposium on Logic in Computer Science (LICS)*, pages 332–344. IEEE Computer Society Press, Cambridge, USA, 1986.

[Vardi and Wolper, 1994] M. Vardi and P. Wolper. Reasoning about Infinite Computations. *Information and Computation*, 115:1–37, 1994.

[Vardi, 1982] M. Vardi. The Complexity of Relational Query Languages. In *Proceedings of the Fourteenth ACM Symposium on Theory of Computing (STOC)*, pages 137–145, 1982.

[Vardi, 1988] M. Y. Vardi. A Temporal Fixpoint Calculus. In *Proceedings of the Fifteenth Annual ACM Symposium on Principles of Programming Languages (POPL)*, pages 250–259, San Diego, USA, 1988.

[Vardi, 1994] M. Vardi. Nontraditional Applications of Automata Theory. In *Proceedings of International Symposium on the Theoretical Aspects of Computer Software (STACS)*, volume 789 of *LNCS*, pages 575–597. Springer-Verlag, 1994.

[Vardi, 1996] M. Vardi. An Automata-Theoretic Approach to Linear Temporal Logic. In F. Moller and G. Birtwistle, editors, *Logics for Concurrency*, pages 238–266. Springer Verlag, 1996.

[Vazirgiannis and Wolfson, 2001] M. Vazirgiannis and O. Wolfson. A Spatiotemporal Model and Language for Moving Objects on Road Networks. In *Proceedings of the International Symposium on Advances in Spatial and Temporal Databases (SSTD)*, pages 20–35, 2001.

[Veloso *et al.*, 1990] M. Veloso, A. Pérez, and J. Carbonell. Non-Linear Planning with Parallel Resource Allocation. In *Proceedings of the DARPA Workshop on Innovative Approaches to Planning, Scheduling and Control*, pages 207–212, 1990.

[Veloso *et al.*, 1995] M. Veloso, J. Carbonell, A. Pérez, D. Borrajo, E. Fink, and J. Blythe. Integrated Planning and Learning: the PRODIGY Architecture. *Journal of Experimental and Theoretical AI*, 7(1), 1995.

[Vendler, 1967] Z. Vendler. *Linguistics and Philosophy*. Cornell University Press, Ithaca, USA, 1967.

[Venema, 1991a] Y. Venema. A Modal Logic for Chopping Intervals. *Journal of Logic and Computation*, 1(4):453–476, 1991.

[Venema, 1991b] Y. Venema. Completeness via Completeness. In M. de Rijke, editor, *Colloquium on Modal Logic, 1991*. ITLI-Network Publication, Instit. for Lang., Logic and Information, University of Amsterdam, Netherlands, 1991.

[Venkatesh, 1986] G. Venkatesh. A Decision Method for Temporal Logic based on Resolution. *Lecture Notes in Computer Science*, 206:272–289, 1986.

[Vere, 1983] S. Vere. Planning in time: Windows and durations for activities and goals. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 5, 1983.

[Vidal and Ghallab, 1996] T. Vidal and M. Ghallab. Constraint-Based Temporal Management in Planning: the IxTeT way. In *Proceedings of the Twelfth European Conference on Artificial Intelligence (ECAI)*, 1996.

[Vila and Reichgelt, 1996] L. Vila and H. Reichgelt. The Token Reification Approach to Temporal Reasoning. *Artificial Intelligence*, 83(1):59–74, May 1996.

[Vila and Schwalb, 1996] L. Vila and E. Schwalb. A Theory of Time and Temporal Incidence based on Instants and Periods. In *Proceedings of the International Workshop on Temporal Representation and Reasoning (TIME)*, pages 21–28. IEEE Computer Society Press, 1996.

[Vila and Yoshino, 1996] L. Vila and H. Yoshino. Time in Automated Legal Reasoning (the long report). Technical Report 96-57, UC Irvine, California, USA, 1996.

[Vila, 1994] L. Vila. *IP*: An Instant-Period–based Theory of Time. In R. Rodriguez, editor, *Proceedings of the ECAI'94 Workshop on Spatial and Temporal Reasoning*, 1994.

[Vilain and Kautz, 1986] M. Vilain and H. Kautz. Constraint Propagation Algorithms for Temporal Reasoning. In *Proceedings of the Fifth (US) National Conference on Artificial Intelligence (AAAI)*, pages 377–382, 1986.

[Vilain *et al.*, 1990] M. Vilain, H.A. Kautz, and P. van Beek. Constraint Propagation Algorithms for Temporal Reasoning: A Revised Report. In D. S Weld and J. de Kleer, editors, *Readings in Qualitative Reasoning about Physical Systems*, pages 373–381, San Mateo, USA, 1990. Morgan Kaufmann.

[Vilain, 1982] M. Vilain. A System for Reasoning About Time. In *Proceedings of the 2nd (US) National Conference on Artificial Intelligence (AAAI '82)*, pages 197–201, Pittsburgh, PA, USA, August 1982. American Association for Artificial Intelligence.

[Visser *et al.*, 2000] W. Visser, K. Havelund, G. Brat, and S. Park. Model Checking Programs. In *Proceedings of International Conference on Automated Software Engineering (ASE)*, September 2000.

[von Wright, 1965] G. H. von Wright. And Next. *Acta Philosophica Fennica*, 18:293–304, 1965.

[Wainer and de Melo Rezende, 1997] J. Wainer and A. de Melo Rezende. A Temporal Extension to the Parsimonious Covering Theory. *Artificial Intelligence in Medicine*, 10:235–255, 1997.

[Wainer and Sandri, 1999] J. Wainer and S. Sandri. Fuzzy Temporal/Categorical Information in Diagnosis. *Journal of Intelligent Information Systems*, 13:9–26, 1999.

[Walker, 1948] A.G. Walker. Durées et Instants. *Revue Scientifique*, 85:131–134, 1948.

[Walther, 1987] C. Walther. *A Many Sorted Calculus Based on Resolution and Paramodulation*. Pitman, 1987.

[Wang *et al.*, 1995] X. S. Wang, S. Jajodia, and V. Subrahmanian. Temporal Modules: An Approach Toward Federated Temporal Databases. *Information sciences*, 82:103–128, 1995.

[Wang *et al.*, 1997] X. S. Wang, C. Bettini, A. Brodsky, and S. Jajodia. Logical Design for Temporal Databases with Multiple Granularities. *ACM Transactions on Database Systems*, 22(2):115–170, 1997.

[Webber, 1995] A. B. Webber. Proof of the Interval Satisfiability Conjecture. *Annals of Mathematics and Artificial Intelligence*, 15, 1995.

[Weida and Litman, 1992] R. Weida and D. Litman. Terminological Reasoning with Constraint Networks and an Application to Plan Recognition. In B. Nebel, W. Swartout, and C. Rich, editors, *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 282–293. Morgan Kaufmann, 1992.

[Weld *et al.*, 1998] D. S. Weld, C. R. Anderson, and D. E. Smith. Extending Graphplan to Handle Uncertainty and Sensing Actions. In *Proceedings of AAAI*, pages 897–904, 1998.

[Weld, 1994] D. S. Weld. An Introduction to Least Commitment Planning. *AI Magazine*, 15(61):27–61, Winter 1994.

[Wetprasit and Sattar, 1998] R. Wetprasit and A. Sattar. Temporal Reasoning with Qualitative and Quantitative Information About Points and Durations. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI)*, pages 656–663, Madison, USA, July 1998.

[Wetprasit *et al.*, 1996] R. Wetprasit, A. Sattar, and L. Khatib. Reasoning with Sequences of Events (an extended abstract). In *Proceedings of the International Workshop on Temporal Representation and Reasoning (TIME)*, pages 36–38. IEEE Computer Society Press, 1996.

[Whitehead, 1919] A.N. Whitehead. *An Enquiry Concerning the Principles of Natural Knowledge*. Cambridge, 1919.

[Widom and Ceri, 1996] J. Widom and S. Ceri, editors. *Active Database Systems: Triggers and Rules for Advanced Database Processing*. Morgan Kaufmann, 1996.

[Wiederhold and Genesereth, 1997] G. Wiederhold and M. Genesereth. The Conceptual Basis of Mediation Services. *IEEE Expert*, 12(5):38–47, 1997.

[Wiederhold, 1992] G. Wiederhold. Mediators in the Architecture of Future Information Systems. *IEEE Computer*, 25(3):38–50, 1992.

[Wijsen, 1998] J. Wijsen. Reasoning about Qualitative Trends in Databases. *Information Systems*, 23(7):463–487, 1998.

[Wijsen, 1999] J. Wijsen. Temporal FDs on Complex Objects. *ACM Transactions on Database Systems*, 24(1):127–176, 1999.

[Wijsen, 2000] J. Wijsen. A String-based Model for Infinite Granularities. In C. Bettini and A. Montanari, editors, *Proceedings of the AAAI Workshop on Spatial and Temporal Granularities*, pages 9–16. AAAI Press, 2000.

[Wilkins, 1988] D.E. Wilkins. *Practical Planning: Extending the Classical AI Planning Paradigm*. Morgan Kaufmann Publishers Inc., San Francisco, CA, 1988.

[Williams, 1986] B. C. Williams. Doing Time: Putting Qualitative Reasoning on Firmer Ground. In *Proceedings of AAAI*, pages 105–112. AAAI press, 1986.

[Wolper, 1983] P. Wolper. Temporal Logic can be More Expressive. *Information and Computation*, 56(1–2):72–99, 1983.

[Wolper, 1985] P. Wolper. The Tableau Method for Temporal Logic: An Overview. *Logique et Analyse*, June–Sept 1985.

[Wolper, 1989] P. Wolper. On the Relation of Programs and Computations to Models of Temporal Logic. In B. Banieqbal, B. Barringer, and A. Pnueli, editors, *Temporal Logic in Specification*, pages 75–123. Springer-Verlag, LNCS 398, 1989.

[Wolter and Zakharyaschev, 1998a] F. Wolter and M. Zakharyaschev. Satisfiability Problem in Description Logics with Modal Operators. In A. G. Cohn, L. Schubert, and S. C. Shapiro, editors, *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 512–523. Morgan Kaufmann, San Francisco, California, 1998.

[Wolter and Zakharyaschev, 1998b] F. Wolter and M. Zakharyaschev. Temporalizing Description Logics. See `citeseer.ist.psu.edu/wolter98temporalizing.html`, 1998.

[Wolter and Zakharyaschev, 1999] F. Wolter and M. Zakharyaschev. Modal Description Logics: Modalizing Roles. *Fundamenta Informaticae*, 39(4):411–438, 1999.

[Wolter, 2000] F. Wolter. The Product of Converse PDL and Polymodal K. *Journal of Logic and Computation*, 10(2):223–251, 2000.

[Wooldridge and Jennings, 1995] M. Wooldridge and N. R. Jennings. Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.

[Wooldridge *et al.*, 2002] M. Wooldridge, M. Fisher, M.-P. Huget, and S. Parsons. Model checking multiagent systems with MABLE. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 952–959, Bologna, Italy, 2002.

[Wooldridge., 1992] M. J. Wooldridge. *The Logical Modelling of Computational Multi-Agent Systems*. PhD thesis, Department of Computation, UMIST, Manchester, UK, 1992.

[Wooldridge, 1996] M. J. Wooldridge. A Knowledge-Theoretic Semantics for Concurrent MetateM. In J. P. Müller, M. J. Wooldridge, and N. R. Jennings, editors, *Intelligent Agents III — Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages (ATAL)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Heidelberg, 1996.

[Wooldridge, 2000] M. Wooldridge. *Reasoning about Rational Agents*. MIT Press, 2000.

[Wooldridge, 2002] M. Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons, 2002.

[Yampratoom and Allen, 1993] E. Yampratoom and J. Allen. Performance of Temporal Reasoning Systems. *SIGART Bulletin*, 4(3):26–29, 1993.

[Yang *et al.*, 2000] J. Yang, H. C. Ying, and J. Widom. TIP: A Temporal Extension to Informix. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, page 596, 2000.

[Yang, 1997] Q. Yang. *Intelligent Planning*. Springer-Verlag, Berlin, Heidelberg, New York, 1997.

[Yi *et al.*, 1997] W. Yi, K.G. Larsen, and P. Pettersson. Uppaal in a Nutshell. *International Journal of Software Tools for Technology Transfer*, 1(1), 1997.

[Yoshino, 1994a] H. Yoshino. Representation of Legal Knowledge by Compound Predicate Formula. In D. Tiscornia C. Biagioli, G. Sartor, editor, *Proceedings of the ICLP Workshop on Legal Application of Logic Programming*, pages 128–137. MIT press, 1994.

[Yoshino, 1994b] H. Yoshino. Representation of Legal Knowledge by Legal Flowchart and Compound Predicate Formula. Technical Report TM-1298, ICOT, 1994.

[Younes and Simmons, 2003] H.L.S. Younes and R.G. Simmons. VHPOP: Versatile Heuristic Partial Order Planner. *Journal of AI Research*, (Special issue on 3rd International Planning Competition), 2003.

[Zeman, 1973] J. Zeman. *Modal logic: The Lewis Modal Systems*. Oxford University Press, 1973.

[Zhang *et al.*, 2002] D. Zhang, V. J. Tsotras, and B. Seeger. Efficient Temporal Join Processing using Indices. In *Proceedings of IEEE International Conference on Data Engineering*, pages 103–114, 2002.

[Zilberstein, 1996] S. Zilberstein. Using anytime algorithms in intelligent systems. *AI magazine*, 17(3):73–83, 1996.