Annals of Mathematics and Artificial Intelligence 22 (1998) 159–192

Dynamic temporal interpretation contexts for temporal abstraction

Yuval Shahar

Section on Medical Informatics, Medical School Office Building (MSOB) x215, Stanford University, 251 Campus Drive, Stanford, CA 94305–5479, USA
E-mail: shahar@smi.stanford.edu

Temporal abstraction is the task of abstracting higher-level concepts from time-stamped data in a context-sensitive manner. We have developed and implemented a formal knowledge-based framework for decomposing and solving that task that supports acquisition, maintenance, reuse, and sharing of temporal-abstraction knowledge. We present the logical model underlying the representation and runtime formation of *interpretation contexts*. Interpretation contexts are relevant for abstraction of time-oriented data and are induced by input data, concluded abstractions, external events, goals of the temporal-abstraction process, and certain combinations of interpretation contexts. Knowledge about interpretation contexts is represented as a *context ontology* and as a *dynamic induction relation* over interpretation contexts and other proposition types. Induced interpretation contexts are either *basic, composite, generalized*, or *nonconvex*. We provide two examples of applying our model using an implemented system; one in the domain of clinical medicine (monitoring of diabetes patients) and one in the domain of traffic engineering (evaluation of traffic-control actions). We discuss several distinct advantages to the explicit separation of interpretation-context propositions from the propositions inducing them and from the abstractions created within them.

1. Introduction: the temporal-abstraction task

Many domains require the collection of substantial numbers of data over time and the abstraction of those data into higher-level concepts, meaningful for that domain. Much work had been done regarding the structure of time and the nature of general temporal reasoning. Our main interest, however, concerns the specific temporal-reasoning task of context-sensitive abstraction and interpretation of time-stamped data.

We will employ examples mainly from the domain of clinical medicine. As will be apparent, the ideas are quite general and are applicable to other time-oriented domains; we include one example (traffic control).

Most clinical tasks require measurement and capture of numerous patient data. Physicians who have to make decisions based on these data may be overwhelmed by the number of data if their ability to *reason* with the data does not scale up to the data-storage capabilities. Most data include a time stamp in which each particular datum was valid; an emerging pattern over a span of time, especially in a specific context (e.g.,

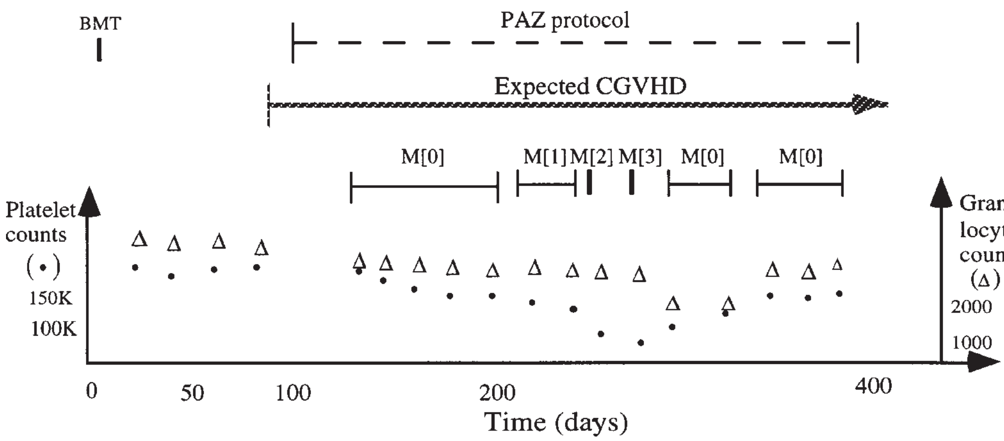


Figure 1. Abstraction of platelet and granulocyte values during administration of the prednisone/azathioprine (PAZ) clinical protocol for treating patients who have chronic graft-versus-host disease (CGVHD). The time line starts with a bone-marrow transplantation (BMT) event. ⊢ — ⊣ = event; • = platelet counts; Δ = granulocyte counts; ⊢→ = open context interval; ⊢⊣ = closed abstraction interval; $M[n]$ = myelotoxicity (bone-marrow-toxicity) grade n.

therapy with a particular drug), has much more significance than an isolated finding or even a set of findings. Thus, it is highly desirable for an automated knowledge-based decision-support tool that assists physicians who monitor patients over significant periods to provide short, informative, context-sensitive summaries of time-oriented clinical data stored on electronic media. Such a tool should be able to answer queries at various levels of abstraction about abstract concepts that summarize the data. Data summaries are valuable to the physician, support an automated system's diagnostic or therapeutic recommendations, and monitor plans suggested by the physician or by the decision-support system. A meaningful summary cannot use only time points, such as data-collection dates; it must be able to characterize significant features over *periods* of time, such as "2 weeks of grade-II bone-marrow toxicity in the context of therapy for potential complications of a bone-marrow transplantation event" (figure 1) and more complex patterns. The temporal-abstraction (TA) task is thus an interpretation task: given time-stamped data and external events, produce context-specific, interval-based, relevant abstractions of the data (a more formal definition will be stated in section 3, when we present the formal ontology – terms and relations – of the TA task).

Temporal abstractions, however, are meaningful only within the temporal span of a relevant context, such as the (possibly delayed) effect of the drug AZT. Contexts create a relevant frame of reference for the interpretation process. On one hand, contexts enable the creation of context-sensitive abstractions (e.g., the meaning of many measurements depends on the context in which they were made); on the other hand, contexts focus the computation process on a limited set of data, for which the abstraction is relevant (e.g., reasoning about complications of bone-marrow transplantations is relevant only if a bone marrow transplantation event occurred in the past, possibly even within a particular time window).

In this paper, we focus on one of the key TA subtasks: formation of appropriate temporal contexts for interpretation of the time-oriented data. In section 3, we define briefly the formal ontology used by all the TA mechanisms. In section 4, we present the context-forming mechanism, which uses that ontology, when mapped to the matching domain knowledge, to create temporal contexts for interpretation of the data in a context-sensitive manner. We also explain the meaning of the distinctions made by the context-forming mechanism among various types of interpretation contexts. In section 5, we present two examples of the application of the KBTA method and in particular of the interpretation-context model to a clinical (diabetes therapy) domain and to an engineering (traffic-control) domain, two of the domains in which the RÉSUMÉ system was employed. In section 6, we relate our approach to several general formalisms for the representation of and reasoning about contexts, and to several somewhat more domain-specific frameworks and systems (in particular, in medical domains) that solve tasks comparable to the TA task. Section 7 concludes with a brief summary and discussion of limitations and advantages of our approach.

2. Knowledge-based temporal abstraction

A method solving the TA task encounters several conceptual and computational problems:

(1) both the input data and the required output abstractions might include several data types (e.g., symbolic, numeric) and can exist at various abstraction levels;

(2) input data might arrive out of temporal order, and existing interpretations must be revised nonmonotonically;

(3) several alternate interpretations might need to be maintained and followed over time;

(4) parameters have context-specific temporal properties, such as expected persistence of measured values and classification functions (e.g., the meaning of the value low of the hemoglobin-state abstraction depends on the context);

(5) acquisition of knowledge from domain experts and maintenance of that knowledge should be facilitated.

The method should enable *reusing* its *domain-independent* knowledge for solving the TA task in other domains, and enable *sharing* of *domain-specific* knowledge with other tasks in the same domain.

The framework that we are using for solving the TA task is based on our work on temporal-abstraction mechanisms [21,22,24,25,27]. We have defined a general *problem-solving method* [6] for interpreting data in time-oriented domains, with clear semantics for both the *method* and its domain-specific *knowledge* requirements: the **knowledge-based temporal-abstraction (KBTA) method**. The KBTA method comprises a knowledge-level representation of the TA task and of the knowledge required

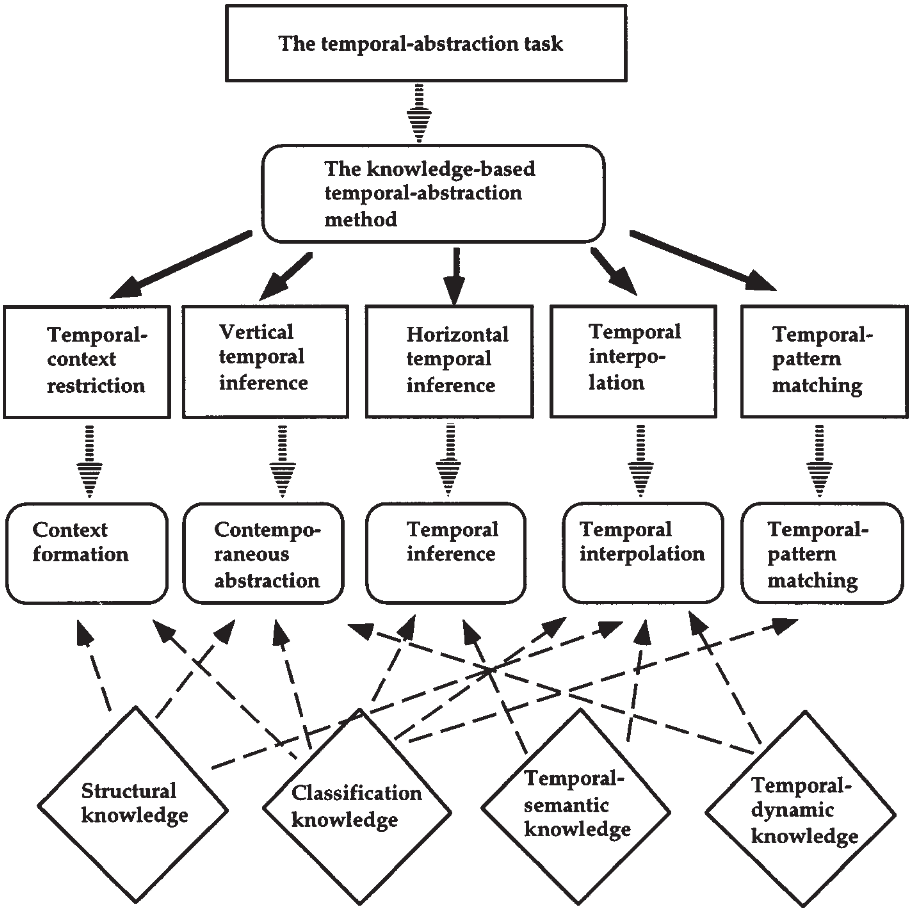


Figure 2. The knowledge-based temporal-abstraction method and its mechanisms. ▭ = task; ⬭ = method or mechanism; ◇ = knowledge type; ➡ = DECOMPOSED-INTO relation; ⇢ = SOLVED-BY relation; - - ➝ = USED-BY relation.

to solve that task. The KBTA method has a formal model of input and output entities, their relations, and their properties – the **KBTA ontology** [21,22].

The KBTA method decomposes the TA task into five parallel *subtasks* (figure 2):

(1) **Temporal-context restriction**: creation of contexts relevant for data interpretation (e.g., effect of a drug), to focus and limit the scope of the inference (we elaborate on this task in section 4).

(2) **Vertical temporal inference**: inference from values of contemporaneous input data or abstractions (e.g., results of several different blood tests conducted during the same day) into values of higher-level concepts (e.g., classification into bone-marrow toxicity Grade II). Note that classification functions are often context sensitive (e.g., different clinical guidelines define differently MODERATE ANEMIA, a state abstraction of the hemoglobin-level clinical parameter).

(3) **Horizontal temporal inference**: inference from similar-type propositions that hold over different time intervals (e.g., joining different-value abstractions of the same parameter that hold over two meeting time intervals and computing the new abstraction's value). In fact, two subtasks are involved: *temporal-semantic inference* infers specific types of interval-based logical conclusions, given interval-based propositions, using a deductive extension of Shoham's temporal semantic properties [28]. For instance, unlike two anemia periods, two episodes of 9-month pregnancies can never be summarized as an episode of an 18-month pregnancy – even if they followed each other – since they are not *concatenable*, a temporal-semantic property. Similarly, a week-long episode of coma implies an abstraction of coma during each day (i.e., it has the *downward-hereditary* temporal-semantic property); that is not necessarily true for the abstraction "a week of oscillating blood pressure." *Temporal horizontal inference* determines the domain value of an abstraction created from two joined abstractions (e.g., for most parameters and interpretation contexts, DECREASING and SAME might be concatenated into NONINCREASING). Although more stable than classification functions, temporal-semantic properties might be sensitive to the context.

(4) **Temporal interpolation**: bridging of gaps between similar-type but temporally disjoint point- or interval-based propositions to create longer intervals (e.g., joining two disjoint episodes of anemia, occurring during different days, into a longer episode, bridging the gap between them). We use *local* (forward and backward from an abstraction interval) and *global* (between two abstraction intervals) *truth-persistence functions* to model a belief in the value of an abstraction [21]. Global truth-persistence (Δ) functions return the maximal temporal-gap threshold that can be bridged (with a high-enough probability) between two temporally disjoint abstractions, given the parameter involved, its value(s), the length of each abstraction, and the interpretation context of the abstractions. Both local and global truth-persistence functions are highly dependent on context.

(5) **Temporal-pattern matching**: creation of intervals by matching patterns over disjoint intervals, over which propositions of various types (including other patterns) hold. For example, in the domain of bone-marrow transplantation (see figure 1), *quiescent-onset chronic graft-versus-host disease (GVHD)* is a pattern-abstraction parameter defined as "*chronic GVHD starting at least 100 days after a bone-marrow transplantation event, but within one month of the end of a preceding acute GVHD*." Temporal patterns might be meaningful only for certain contexts and should not be created (and their creation should not be attempted) in others.

The five subtasks of the KBTA *method* are solved by five **temporal-abstraction mechanisms** (nondecomposable computational modules) that we have defined (see figure 2). The temporal-abstraction mechanisms depend on four well-defined domain-specific **knowledge types**: *structural* knowledge (e.g., IS-A, PART-OF and ABSTRACTED-INTO relations), *classification* (functional) knowledge (e.g., mapping of hemoglobin values into hemoglobin states), *temporal-semantic* (logical) knowledge (e.g., the CONCATENABLE property [28]), and *temporal-dynamic* (probabilistic) knowledge (e.g., temporal persistence functions that bridge gaps between temporally disjoint intervals [21]). Values for the four knowledge types are specified as the domain's **temporal-abstraction ontology** when developing a temporal-abstraction system for a particular domain and task.

We have implemented the KBTA method as the **RÉSUMÉ** system [24] and applied it with encouraging results to several clinical domains such as chronic graft-versus-host disease [24], monitoring of children's growth [14], therapy of AIDS patients [21], and therapy of patients who have diabetes [25]. We also have used the KBTA method and the RÉSUMÉ system to model and solve a spatiotemporal traffic-monitoring task [18,23]. The RÉSUMÉ system is currently used within the **Tzolkin** temporal-mediator server, a part of the **EON** architecture for support of clinical-guideline-based therapy [19].

3. The temporal-abstraction ontology

Informally, the KBTA temporal model includes both time intervals and time points. *Time points* are the basic temporal primitives, but propositions, such as occurrence of events and existence of parameter values, can be interpreted only over *time intervals*. Therefore, all propositions are *fluents* [17] and, in our model, must be interpreted over a particular time *period* (e.g., the value of the temperature parameter during time interval $[t_1, t_2]$). The knowledge-based TA ontology [22] contains the following entities:

1. *Time stamps*, $\tau_i \in T$, comprise the basic primitives of time. A time-standardization function, $f_S(\tau_i)$, can map a time stamp into an integer amount of any pre-defined temporal granularity unit $G_i \in \Gamma$ (e.g., hour). Time stamps are measured in G_i units with respect to a zero-point time stamp. There is a time unit G_0 of the lowest granularity. A finite positive or negative amount of G_i units is a *time measure*. There is a total order on time stamps. Subtraction of any time stamp from another must be defined and should return a time measure. Addition or subtraction of a time measure to or from a time stamp must return a time stamp.

2. A *time interval* is an ordered pair of time stamps that denote the endpoints, [*I*.start, *I*.end], of the interval. A zero length interval in which *I*.start $=$ *I*.end is a time point.

3. An *interpretation context* $\xi \in \Xi$ is a proposition representing a state of affairs relevant to interpretation (e.g., "the drug insulin exerts its effect during this interval"). When an interpretation context exists during a particular time interval, parameters may be interpreted differently within that time interval. IS-A and SUBCONTEXT relations are defined over the set of interpretation contexts. *Basic* interpretation contexts are atomic propositions. *Composite interpretation contexts* are created

by the conjunction of a basic or a composite interpretation context and one of its subcontexts. Intuitively, composite interpretation contexts permit the definition of a hierarchy of increasingly specific contexts. Unlike the propositions inducing them, interpretation contexts are assumed as a default to have the *concatenable* [28] temporal-semantic property. *Generalized* and *nonconvex interpretation contexts* are special types of interpretation contexts (see sections 4.2 and 4.3).

4. A *context interval* is a structure $\langle \xi, I \rangle$ containing an interpretation context ξ and a time interval I (i.e., an interpretation context during an interval).

5. An *event proposition* or event $e \in E$ is the occurrence of an external willful act or process, such as the administration of a drug. Events are instantiated event schemata; an event schema has a series a_i of event attributes (e.g., drug dose) that must be mapped to attribute values ν_i. A PART-OF (or *subevent*) relation is defined over event schemata.

6. An *event interval* is a structure $\langle e, I \rangle$, consisting of an event proposition e and a time interval I that represents the duration of the event.

7. A parameter schema or *parameter* $\pi \in \Pi$ is a measurable or describable state of the world. Parameters may represent raw input data (e.g., a hemoglobin level) or abstractions from the raw data (e.g., a state of anemia). Parameter schemata have various *properties*, such as a domain V_π of possible symbolic or numeric values, measurement units, temporal-semantic properties, or temporal persistence. An *extended parameter* is a combination $\langle \pi, \xi \rangle$ of a parameter π and an interpretation context ξ. An extended parameter is also a parameter and can have properties. Extended parameters have a special property, a value $\nu \in V_\pi$, which is typically known only at runtime (i.e., parameter values require a context). A *parameter proposition* is the combination of a parameter, a parameter value, and an interpretation context, $\langle \pi, \nu, \xi \rangle$ (e.g., "the state of hemoglobin is low in the context of chemotherapy").

8. A *parameter interval* $\langle \pi, \nu, \xi, I \rangle$ is a parameter proposition and a time interval, representing the value of a parameter in a specific context during a particular time interval.

9. An *abstraction function* $\theta \in \Theta$ is a unary or multiple-argument function from one or more parameters to an abstract parameter. The abstract parameter has one of three abstraction types: *state*, *gradient*, and *rate*. An additional type of abstraction is *pattern* which defines a temporal pattern of several other parameters. An abstraction of a parameter is a parameter (thus, both hemoglobin and the state of hemoglobin are parameters, with distinct properties).

10. An *abstraction* is a parameter interval $\langle \pi, \nu, \xi, I \rangle$ where π is an abstract parameter. Abstractions may be abstraction points or abstraction intervals.

11. An *abstraction goal* $\psi \in \Psi$ is a proposition that indicates a goal or intention that is relevant to the TA task (e.g., the intention to control a diabetes patient's blood-glucose values).

12. An *abstraction-goal interval* is a structure $\langle \psi, I \rangle$, where ψ is a temporal-abstraction goal that is posted during the interval I. An abstraction-goal interval induces interpretation contexts.

13. Interpretation contexts are dynamically *induced* by (or inferred from) event, parameter, or abstraction-goal propositions. The time intervals over which the inducing propositions exist impose temporal constraints on the interval in which the inferred context will be valid. For example, the interpretation context of insulin's effect on blood-glucose values might begin at least 30 minutes following the event of insulin administration and end up to 8 hours after terminating the administration. These constraints are represented formally in a *dynamic induction relation of a context interval* (DIRC). A DIRC is a relation over propositions and time measures, in which each member is a structure of the form $\langle \xi, \varphi, ss, se, es, ee \rangle$. Intuitively, the inducing proposition is assumed, at runtime, to be interpreted over some time interval I with known end points. The symbol ξ is the induced interpretation context. The symbol $\varphi \in P$ represents the inducing proposition, an event, an abstraction-goal, or a parameter proposition. Each of the other four symbols is either the "wild card" symbol *, or a time measure, which denote, respectively, the temporal distance between the start point of I and the start point of the induced context interval, the distance between the start point of I and the end point of the induced context interval, the distance between the end point of I and the start point of the context interval, and the distance between the end point of I and the end point of the induced context interval (figure 3). Note that the resultant context interval need not span the same temporal scope as the inducing proposition, but can have any of Allen's 13 relations to it [1] (see figure 3b). A *context-forming proposition* is an inducing proposition in at least one DIRC.

A TA ontology of a domain is an event ontology, a context ontology, a parameter ontology, a set of abstraction-goal propositions, and the set of all DIRCs for a particular domain. The event ontology of a domain consists of the set of all the relevant event schemata and propositions. The context ontology defines the set of all the relevant contexts and subcontexts. The parameter ontology is composed of the set of all the relevant parameter propositions and their properties. The TA task also assumes the existence of a set of temporal queries, expressed in a predefined temporal-abstraction language. A *temporal query* is a set of temporal and value constraints over the components of a set of parameter and context intervals [21].

The TA task solved by the KBTA method is thus the following: given at least one abstraction-goal interval, a set of event intervals, a set of parameter intervals, and the domain's TA ontology, produce an interpretation – that is, a set of context intervals and a set of new abstractions such that the interpretation can answer any temporal

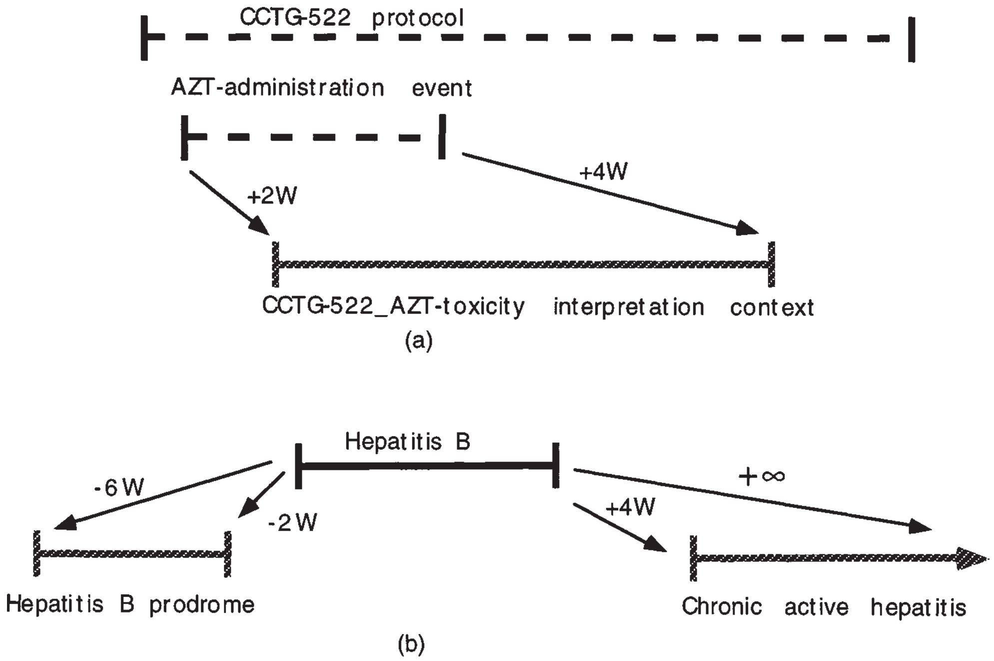


Figure 3. Dynamic induction relations of context intervals (DIRCs). (a) An AZT-toxicity interpretation context induced by an AZT-administration event within the context of a CCTG-522 AIDS-therapy experimental-protocol event (which induces a contemporaneous CCTG-522 context interval). The AZT-toxicity interpretation context starts 2 weeks after the start of the inducing AZT-administration event, and ends 4 weeks after its end. In this case, the result is a composite (nested) interpretation context, CCTG-522_AZT-toxicity, since AZT-toxicity is a subcontext of CCTG-522 in the context ontology (see section 4). (b) Prospective (chronic active hepatitis) and retrospective (hepatitis B prodrome) interpretation contexts induced by the hepatitis B parameter proposition. ⊢ — ⊣ = event interval; ⊢⊣ = closed context interval; ⊢→ = open context interval; ⊢⊣ = closed abstraction interval.

query about all the abstractions derivable from the transitive closure of the input data and the domain's TA ontology.

The four types of domain-specific knowledge required by the TA mechanisms, apart from the event and context ontologies, are represented in the RÉSUMÉ system in the **parameter-properties ontology**, a representation of the parameter ontology [24]. The parameter-properties ontology is a frame hierarchy that represents knowledge about parameter propositions (e.g., classification knowledge about various values of the hemoglobin state abstraction) and specializes that knowledge within different interpretation contexts (e.g., therapy by different drugs). Figure 4 shows a part of the RÉSUMÉ parameter-properties ontology in the domain of protocol-based care.

Note that the knowledge regarding the granulocyte-state abstract parameter is specialized in the context of the CCTG-522 experimental protocol for therapy of patients who have AIDS, in the context of the prednisone/azathioprine (PAZ) experimental protocol for treating chronic graft-versus-host disease, and in the subcontext of each part of the PAZ protocol.

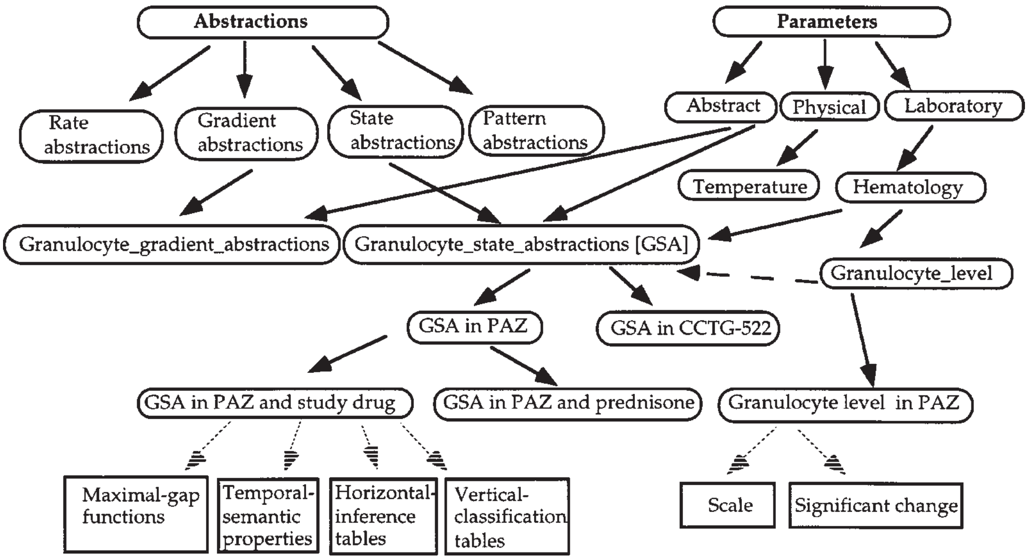


Figure 4. A portion of the RÉSUMÉ parameter-properties ontology in the domain of protocol-based care, showing a specialization of the temporal-abstraction properties for the granulocyte_state_abstraction (GSA) abstract parameter in the context of the prednisone/azathioprine (PAZ) experimental protocol for treating chronic graft-versus-host disease, and in the context of each part of that protocol. = class; = property; = IS-A relation; = PROPERTY-OF relation. = ABSTRACTED-INTO relation.

4. Dynamic induction of context intervals

Abstractions are meaningful only within the span of a relevant context interval, such as administration of the drug AZT as part of a particular clinical protocol for therapy of AIDS. Context intervals create a relevant frame of reference for interpretation, and thus enable a TA mechanism to conclude abstractions for – and only for – that context. Thus, interpretation contexts enable creation of context-sensitive abstractions, while focusing the computation process on a highly limited set of data.

Context intervals are created by the **context-forming mechanism**. The input of the context-forming mechanism is a set of event intervals, parameter intervals, and abstraction goals, and the domain's TA ontology (in particular, the context ontology and the set of DIRCs). The output of the mechanism is a set of context intervals induced by the transitive closure of the input data and the domain-specific knowledge.

As explained in section 3, DIRCs represent relationships between context intervals and several types of propositions that can induce them. Context intervals might be induced by the existence of an *abstraction-goal interval*, such as "therapy of insulin-dependent diabetes," or by the existence of an *event interval*, that is, an external process or action, such as treatment in accordance with a particular clinical protocol. A context interval can also be induced by the existence of a *parameter interval* that includes a *context-forming* (see section 3) parameter proposition $\langle \pi, \nu, \xi \rangle$ – namely, the value ν of the parameter π, in the context ξ, is sufficiently important to change the frame of reference for one or more other parameters (e.g., the LOW value of the hemoglobin-state abstract parameter in the context of protocol CCTG-522 might affect the interpretation of values of the platelet-value parameter).

A *composite interpretation context* (see section 3) can be composed by the context-forming mechanism at runtime from a conjunction of two or more concluded basic interpretation contexts that hold contemporaneously, such that basic context ξ_{i+1} has a SUBCONTEXT relation to basic context ξ_i. The composite interpretation context would be interpreted over a context interval formed from a *temporal intersection* of the two or more corresponding context intervals. For example, components of a composite interpretation context are often induced by an *event chain* – a connected series of events $\langle e_1, e_2, \ldots, e_n \rangle$, where e_{i+1} is a subevent of e_i. In that case, the composite interpretation context would denote an interpretation context induced by the most specific subevent, such as administration of a particular drug as part of a certain protocol. (Subevents of an event often induce interpretation contexts that have a SUBCONTEXT relation to the interpretation context induced by the event.) This knowledge is used as a default in the context ontology, and can also be exploited during a manual or automated process of acquisition of knowledge, either for knowledge elicitation or for knowledge verification and cross-validation. Interpretation contexts can be extended by concatenating two *meeting* [1] equal-context intervals, since they are assumed by default to be concatenable [28], unlike the propositions inducing them, which do not necessarily have that property.

Dynamic induction of context intervals by parameter propositions might lead to new interpretations of existing parameter intervals, and thus might induce new context intervals, within which another parameter proposition, or even the original inducing parameter proposition might have new interpretations (e.g., another state abstraction of the same parameter during the same time, but with a different value). However, we can prove [21] that no contradictions or infinite loops can be generated by the context-forming process.

Claim 1 (*consistency*). The context-forming process has no "oscillation cycles" among different interpretations of the same parameter (i.e., the same parameter proposition can never be retracted and eventually reasserted). That is, no inherent inconsistencies can be created at any point.

Justification. Parameter propositions are not retracted by the addition of a new *interpretation context*. Rather, a new abstraction specific to the new interpretation context is simply added to the set of true parameter propositions. (Retractions *can* occur due to the nonmonotonic nature of temporal abstraction, but in different circumstances, such as arrival of new *data* with a present transaction time but with an old valid time, which might conflict with previous nonmonotonic conclusions, forcing a *view update* [21].) Thus, if a parameter proposition $\langle \pi, \nu_1, \xi_1 \rangle$ holds over time interval I and either that proposition or another one induces a new interpretation context ξ_2 over the same interval I, and within the scope of the new context interval the parameter π is interpreted to have another value ν_2, a new parameter proposition $\langle \pi, \nu_2, \xi_2 \rangle$ would be inferred and added to the set of true propositions. This, of course, creates no contradictions since the parameter π – or, more typically, an abstraction of π, such as state(π) – is interpreted (concurrently) within two different contexts and can thus have two different values at the same time. □

Claim 2 (*stability*). The context-forming process is finite.

Justification. The total number of different interpretation contexts that, potentially, can be inferred (including composite ones) is limited by the number of runtime input data intervals of al types and by a predefined upper bound: the size of the context ontology and the number and depth (in the context-ontology tree) of potential *subcontext chains* (which can form composite contexts) of interpretation contexts that have SUBCONTEXT relations. Furthermore, for each parameter π, the number of possible induced context intervals is bound by the number of DIRCs in which a parameter proposition including π is an inducing proposition.

The process is finite also even if we allow for any (finite) amount of DIRCs that represent induction of prospective and retrospective context intervals. Assume that a parameter proposition induces context interval prospectively (or retrospectively); the new context, together with certain parameter propositions that happen to exist within its scope (and an appropriate DIRC), might induce another prospective or retrospective context interval, etc. The process, however, can only continue for a finite number of steps, since it is limited by both the number of input parameter propositions and the number of DIRCs in the TA ontology of the domain. That is, there cannot be, for instance, an infinite "flip flop" loop between past (retrospective) and future (prospective) context intervals: if the induced contexts affect the interpretation of the same data, they are subject to limitations imposed by the finite nature of the context ontology and the number of DIRCs; if they affect the interpretation of other data, they are subject to the finite size of the input.

Since claim 1 ascertained that there are no retraction/assertion loops (due to inconsistencies) either, the process must end for any finite number of input propositions. □

4.1. *Advantages of explicit contexts and DIRCs*

Explicit interpretation contexts, separate from the propositions inducing them and from abstractions using them, have significant conceptual and computational advantages for context-specific interpretation of time-stamped data.

1. Since the four temporal measures of a DIRC, representing temporal constraints over an induced context interval with respect to the start time and the end time of the inducing proposition, can be positive, negative, or infinite time measures, the context interval induced by a context-forming proposition can have any one of Allen's [1] 13 binary temporal relations (e.g., BEFORE, AFTER, or OVERLAPS) to the time

interval over which the inducing proposition is interpreted (see figure 3). Thus, a context-forming proposition interval can create, in addition to a **direct** (concurrent) context interval, **retrospective** context intervals (e.g., that allows for a potential interpretation of past data as preceding symptoms of a disease), **prospective** context intervals (e.g., that allows for a potential interpretation of future data as complications of a disease), or both (see figure 3). Intuitively, retrospective interpretation contexts represent a form of *abductive* reasoning (e.g., from effects to causes, such as preceding events), while prospective interpretation contexts represent a form of **deductive** reasoning (e.g., from an event to potential complications), or **foresight**. (Note, however, that we infer only an *interpretation context*, which has the potential for allowing a TA mechanism to create a new abstraction, but not the *abstraction* itself.) The context-forming mechanism creates retrospective and prospective contexts mainly to enable the use of context-specific TA functions, such as the correct mapping functions related to ABSTRACTED-INTO relations and the relevant temporal-persistence functions [21], that should not be considered in other contexts. Creation of explicit contexts enables the TA mechanisms to focus on the abstractions appropriate for particular contexts, such as potential consequences of a certain event, and to avoid unnecessary computations in other contexts. In addition, the ability to create dynamically retrospective contexts enables a form of **hindsight** [20], since the interpretation of *present* data can induce new interpretation contexts for the past and thus shed new light on *old* data. Note that the representations of both hindsight and foresight are outside of the scope of formalisms intended for projection or simulation tasks, such as the event calculus [13], in which, in effect, events must directly and instantaneously create state transitions. No delay is permitted for future effects, and past effects are impossible. In our formalism, which geared more towards interpretation, a proposition cannot create directly another proposition, but can induce, over any temporal interval relative to the proposition's temporal scope, an environment (i.e., a context) which enables a TA mechanism to infer another proposition, if the appropriate TA knowledge has been defined for that context (e.g., classification knowledge, which has a functional nature, or temporal-semantic knowledge, which uses logical axioms).

2. Since a context-forming proposition can be an inducing proposition in more than one DIRC, the *same proposition* can *induce* dynamically *several interpretation contexts*, either in the past, the present, or the future, relative to the temporal scope of the interval over which it is interpreted. Thus, we can model, for instance, several potential effects of the same action, each of which creates a different interpretation context, or several inferences from the same temporal pattern, once detected.

3. The *same interpretation context* (e.g., potential bone-marrow toxicity) might be induced by *different propositions*, possibly even of different types and occurring over different periods (e.g., different types of chemotherapy and radiotherapy events). The domain's TA ontology would then be representing the fact that, within the particular interpretation context induced by any of these propositions (perhaps with

different temporal constraints for each proposition), certain parameters would be interpreted in the same way (e.g., we can represent the properties of the hemoglobin-state parameter within the scope of a bone-marrow-toxicity context interval, without the need to list all the events that can lead to the creation of such a context interval). Thus, the separation of interpretation contexts from their inducing propositions also facilitates maintenance and reusability of the TA knowledge base. As mentioned in section 2 when discussing the conceptual requirements of the TA task, facilitation of acquisition and maintenance of TA knowledge has been a major goal in the creation of this framework.

4. Since several context intervals, during which different interpretation contexts hold, can exist contemporaneously, it is possible to represent several abstraction intervals in which the *same abstract parameter* (e.g., the state of the hemoglobin level) has *different values* at the *same time* – one for each valid and relevant context (e.g., "LOW hemoglobin state" in the context of having AIDS without complications, and "NORMAL hemoglobin state" in the context of being treated by the drug AZT, which has expected side effects). Thus, the context-forming mechanism supports maintenance of several *concurrent* **views** of the abstractions in the abstraction database, denoting several possible interpretations of the same data. This is one of the reasons that parameter propositions (including temporal-pattern queries to the abstraction database) must include an interpretation context: the parameter value alone might otherwise be meaningless.

4.2. Generalized interpretation contexts

Additional distinctions important for the TA task are enabled by the explicit use of interpretation contexts and DIRCs. A **simple interpretation context** is a *basic* or a *composite* interpretation context. Our discussion till now concerned simple interpretation contexts. Usually, abstractions are specific to a particular simple interpretation context, and *cannot* be joined (by the temporal-inference or temporal-interpolation mechanisms) to abstractions in other interpretation contexts (e.g., two "LOW hemoglobin state" abstractions might denote different ranges in two different subcontexts of the same interpretation context induced by a chemotherapy-protocol event). This restriction is reasonable, since the primary reason for having contexts is to *limit* the scope of reasoning and of the applicability of certain types of knowledge.

However, it is both desirable and possible to denote that, for certain classes of parameters, contexts, and subcontexts, the abstractions are **sharable** among two meeting different context intervals (i.e., with different interpretation contexts). Such abstractions denote the same state, with respect to the task-related implications of the state, in all sharing contexts. For instance, two meeting "LOW hemoglobin state" abstractions in two different contexts might indeed denote different ranges in the two contexts, and the hemoglobin-state parameter might even have only two possible values in one context, and three in the other, but the domain expert still might want to express the fact that the LOW value of the hemoglobin-state abstraction can be joined meaningfully to summarize a particular hematological state of the patient during the joined time period. The sharable abstraction values would then be defined within a new **generalized interpretation context** that is equivalent to neither of the two shared subcontexts (e.g., those induced by two different parts of the same clinical protocol), nor to their parent context (e.g., the one induced by the clinical protocol itself, within which the hemoglobin-state parameter might have yet another, default, low hemoglobin-state range). This generalized context can be viewed as a generalization of two or more subcontexts of the parent interpretation context. The proposition "LOW hemoglobin-state (within the generalized context)" would then have the logical *concatenable* [28] property and can thus be joined across the temporal scope of two different subcontexts.

4.3. Nonconvex interpretation contexts

Sometimes, we might want to abstract the state of a parameter such as glucose in the preprandial (before meals) interpretation context, over two or more temporally *disjoint*, but semantically equivalent, preprandial interpretation contexts (e.g., the PRELUNCH and PRESUPPER interpretation contexts are both PREPRANDIAL interpretation contexts) (see section 5.1). We might even want to create such an abstraction within only a particular preprandial context (e.g., several PRESUPPER interpretation contexts) skipping intermediate preprandial contexts (e.g., PREBREAKFAST and PRELUNCH interpretation contexts). This interpolation is different from sharing abstractions in a generalized interpretation context, since the abstractions in this case were created within the *same* interpretation contexts, but the interpolation operation joining them needs to skip temporal gaps, including possibly context intervals over which different interpretation contexts hold. The output is a new type of a parameter interval, with respect to temporal scope – a *nonconvex interval*, as defined by Ladkin [15]. A "LOW glucose state" abstraction would be defined, therefore, within the **nonconvex interpretation context** of "prebreakfast episodes". Note that parameter propositions including such a nonconvex context will have different temporal-semantic inference properties [21] from the same parameter propositions except for a simple, convex, context. For instance, propositions will usually not be *downward hereditary* [28] in the usual sense of that property (i.e., the proposition holds within any subinterval of the original interval) unless subintervals are confined to only the convex or nonconvex intervals that the nonconvex superinterval comprises (e.g., only morning times).

Thus, the interpretation context of a parameter proposition is a combination of simple, generalized, and nonconvex interpretation contexts. Assume that a Gen (generalize) operator returns the generalizing-context parent (if it exists) of a parameter proposition in the parameter-properties ontology. Assume that a Gen* operator, that generalizes the Gen operator, returns the least common generalizing-context ancestor (if it exists) $\langle \pi, \nu, \xi_g \rangle$ of two parameter propositions $\langle \pi, \nu, \xi_1 \rangle$, $\langle \pi, \nu, \xi_2 \rangle$, in which the parameter π and the value ν are the same, but the interpretation context is different. Assume that an NC (nonconvex) operator returns the nonconvex-context extension (if it exists) of a parameter proposition. Then, the parameter proposition that represents the nonconvex join (over disjoint temporal spans) of two parameter propositions in which only the interpretation context is different can be represented as

$$\mathrm{NC}\big(\mathrm{Gen}^*\big(\langle\pi,\nu,\xi_1\rangle,\langle\pi,\nu,\xi_2\rangle\big)\big).$$

Thus, we first look for a generalizing interpretation context for glucose-state abstractions in the PRELUNCH and PRESUPPER interpretation contexts, in this case the PREPRANDIAL one. Then we represent the parameter proposition "LOW preprandial glucose-state values" as the LOW value of the glucose-state parameter in the nonconvex extension of the PREPRANDIAL interpretation context. This proposition would be interpreted over some time interval to form a (nonconvex) parameter interval. (Generalized and nonconvex interpretation contexts belong to the context ontology; the corresponding extended-parameter propositions belong to the parameter ontology.)

4.4. Implementation notes: efficiency and complexity

The parameter-properties ontology of the RÉSUMÉ system does not contain a parameter-properties class specialized for *every* interpretation context. A missing specialization in some context signifies that, for that particular context, the abstraction is not relevant (i.e., should not be created for the application), thereby cutting down on unnecessary inferences. Furthermore, each particular TA application in the same domain need not contain instances of every class in the parameter-properties ontology, thus effectively cutting down on possible inferences. In addition, the RÉSUMÉ system enables the designer to prespecify for any application what are the types of desired output parameters (e.g., gradient abstractions) that should be abstracted and which TA mechanisms (e.g., temporal interpolation) should be used [24]. Furthermore, in the EON architecture for guideline-based medical care [19], the RÉSUMÉ system is embedded within a temporal-mediator module, Tzolkin, which answers temporal-abstraction queries by analyzing these queries, then loading from the patient database and from the knowledge base only input data and TA knowledge, respectively, that are potentially relevant to the query. For instance, if certain bone-marrow toxicity abstractions within the PAZ context are required, only data and knowledge within the abstraction tree defined by the (recursive) ABSTRACTED-FROM relation and rooted at the bone-marrow-toxicity class need to be retrieved (e.g., granulocyte-value parameter intervals and granulocyte-state TA properties) as well as events that can induce the PAZ context. (If a parameter proposition is an inducing proposition for a relevant context, the process repeats itself recursively, since the parameter proposition might be abstracted from other parameters, perhaps even in some other context.) Thus, a combination of structural constraints and goal-oriented queries reduces the amount of abstractions generated by RÉSUMÉ, which by itself operates mainly in a data-driven mode. Indeed, operating the RÉSUMÉ system in a pure data-driven mode to produce all possible intermediate and top-level abstractions (which is often useful for certain applications, such as for visualization of the time-oriented data at several abstraction levels, while enabling the user to navigate among all levels) is quite slow. Thus, a delicate balance needs to be struck between highly efficient, but narrow-scoped, goal-driven queries (which might lead to inefficiency when repeated over time), and the slow full data-driven process, which leads to broader-scoped conclusions and which automatically caches intermediate patterns for additional queries (e.g., during a consultation session).

Polynomial limits exist on the complexity of both horizontal classification and temporal interpolation [22]. Temporal-pattern matching can be, in the worst case, exponential in the number of parameter propositions, event propositions, and context intervals. In practice, most patterns involve constraints on only two or three intervals, and the pattern matching is highly constrained by the specified parameter types (e.g., hemoglobin state abstractions), implying essentially a polynomial complexity, albeit with a potentially large constant.

Selective retraction of outdated nonmonotonic conclusions when new data arrive (often with a past time stamp) or when inconsistencies are detected is handled efficiently by an underlying truth-maintenance system [21,24].

5. Application of interpretation contexts

We have applied the RÉSUMÉ system to several different domains. To emphasize the generality of the framework presented in this paper and to give the reader the flavor of the use of dynamic interpretation contexts in some of these applications, we present briefly two examples, one taken from an evaluation of the KBTA framework and the RÉSUMÉ system in a clinical domain (therapy of patients who have insulin-dependent diabetes) and one from an application of the KBTA method to an engineering domain (evaluation of traffic-control actions).

5.1. Interpretation contexts in management of diabetes patients

We have evaluated the RÉSUMÉ system in the domain of monitoring the therapy of patients who have insulin-dependent diabetes mellitus (DM) [25]. We collaborated with two endocrinologists, acquiring within several meetings a TA ontology from one of the experts. We created a parameter-properties ontology for the domain of insulin-dependent diabetes (figure 5), an event ontology (e.g., insulin therapy, meals, physical exercise) (figure 6), and a context ontology (e.g., preprandial [measured at fasting time, before a meal] and postprandial [after a meal] contexts and subcontexts, and postexercise contexts) (figure 7).

In the diabetes-therapy ontology, administrations of regular insulin and of isophane insulin suspension (NPH) are *events* (see figure 6), inducing different insulin-action *interpretation contexts* that are *subcontexts* of the DM *interpretation context* (see figure 8(a)) which represents the context of treating diabetes. Meals are events that induce preprandial and postprandial contexts (see figure 8(b)). Thus, values for

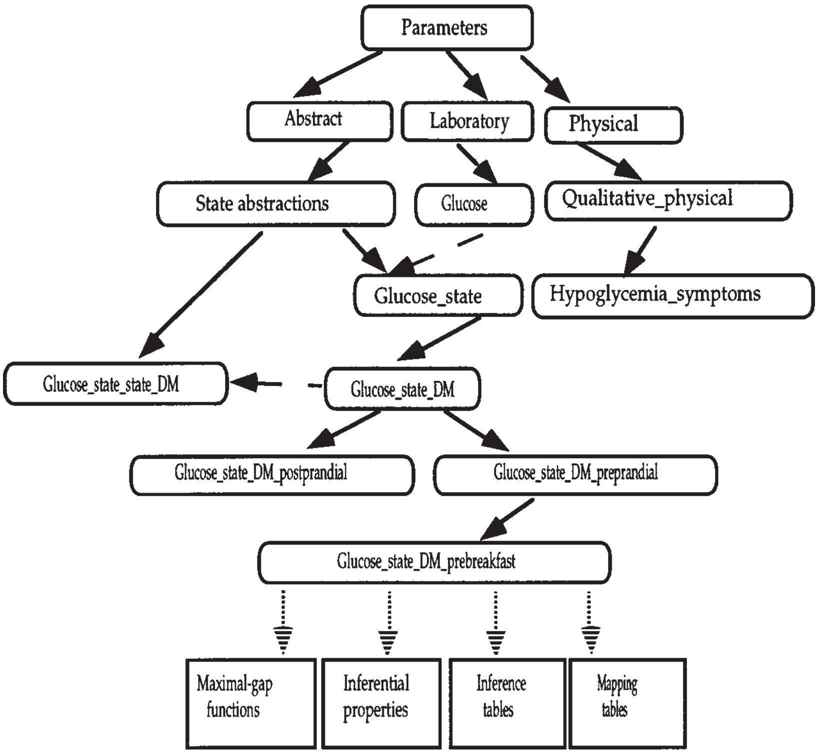


Figure 5. Part of the diabetes parameter-properties ontology. The Glucose parameter is abstracted into the Glucose_state parameter. This abstract parameter has a specialized subclass in the DM context, and is abstracted in that context into the Glucose_state_state parameter. The Glucose_state_DM class is further specialized in the preprandial and postprandial contexts, each of which has several subclasses corresponding to the different relevant premeal contexts. ⬭ = class; ▭ = property; ⟶ = IS-A relation; - - ⟶ = ABSTRACTED-INTO relation; ⦙⦙⦙ = PROPERTY-OF relation; DM = diabetes mellitus.

the Glucuse_state_DM_prebreakfast (the state of glucose in the context of DM and measurement before breakfast) parameter (see figure 5) can be created, when relevant, regardless of absolute time.

The Glucose_state parameter is a new parameter with six values defined from corresponding ranges used by the domain expert (HYPOGLYCEMIA, LOW, NORMAL, HIGH, VERY HIGH, EXTREMELY HIGH). These values are sensitive to the context in which they are generated; for instance, postprandial values allow for a higher range of the normal value. Glucose_state propositions (for all allowed values) have the value TRUE for the temporal-semantic property *concatenable* (see section 2) in the same meal-phase context. The Glucose_state_state parameter is a higher-level abstraction of the Glucose_state parameter, which maps its six values into three (LOW, NORMAL, HIGH, or L, N, H for short). It has different semantic properties, and allows creation of daily horizontal-inference patterns within a *nonconvex* preprandial context (see section 4.3) representing abstraction over several meal phases, such as LLH (LOW, LOW and HIGH

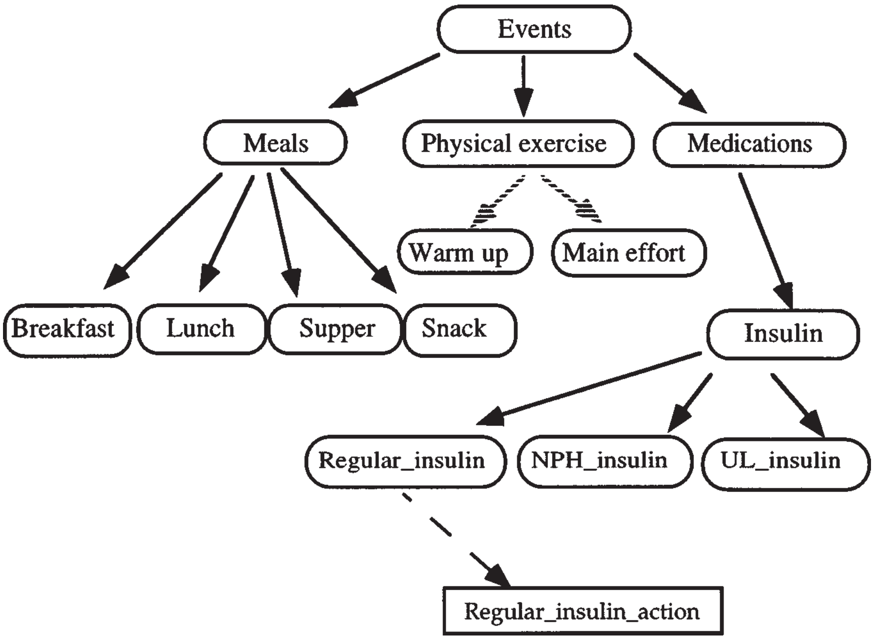


Figure 6. Part of the diabetes event ontology. = class; = induced interpretation context; = IS-A relation; = PART-OF relation; = INDUCED-BY relation.

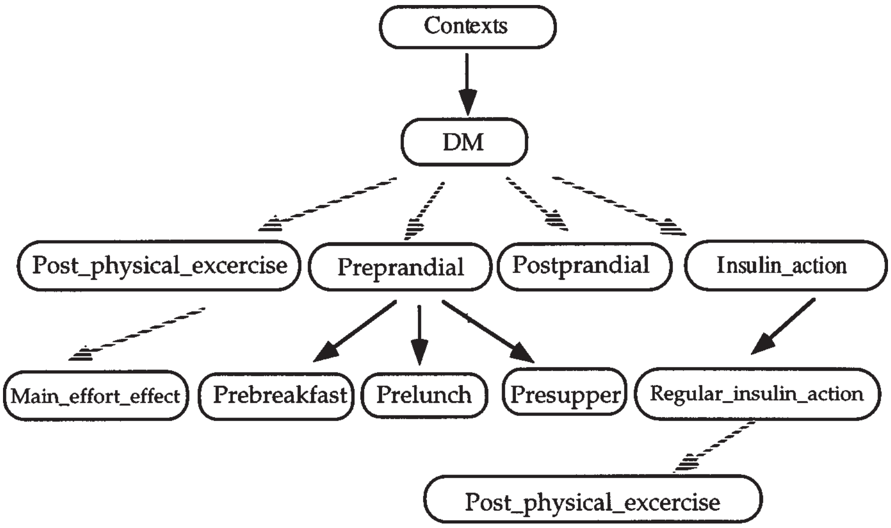


Figure 7. Part of the context ontology in the diabetes-therapy domain. = class; = IS-A relation; = SUBCONTEXT relation; DM = diabetes therapy context. Preprandial and postprandial contexts are induced before and after meal events, respectively. The post-physical-exercise interpretation context has a subcontext relationship to both the DM context and the regular-insulin-action context.

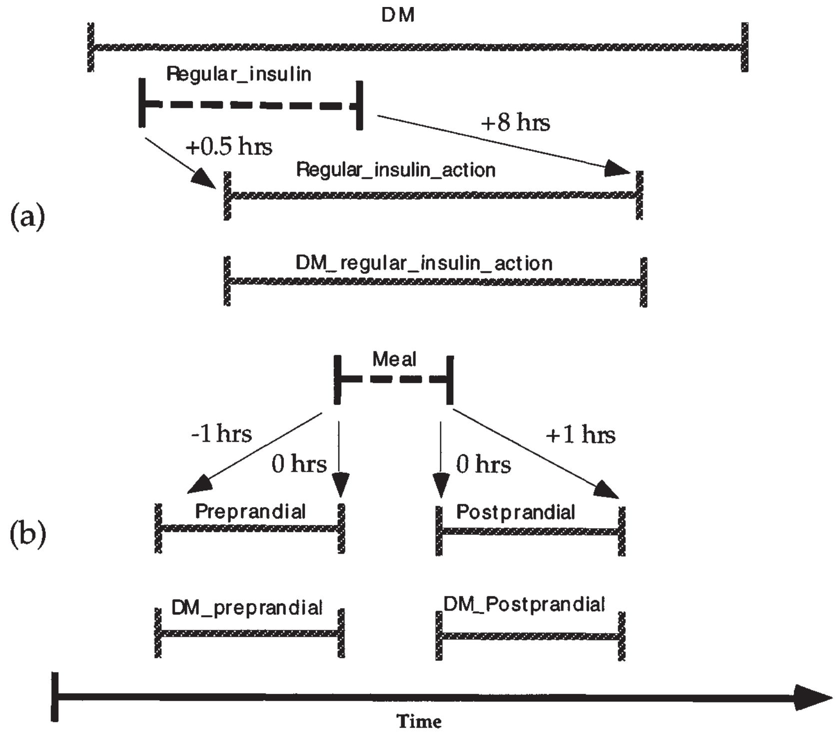


Figure 8. Formation of contexts in the diabetes-treatment domain. (a) Creation of a Regular_insulin_action context, induced by a Regular_insulin administration event, and of the corresponding DM subcontext. (b) Creation of the Postprandial and Preprandial (prospective and retrospective, respectively) context intervals, induced by a Meal event, and formation of the corresponding DM subcontexts. ⊢ — ⊣ = event; ⊢---⊣ = closed context interval. DM = diabetes mellitus (therapy context).

Glucose_state_state values over breakfast, lunch, and supper, respectively). Patterns such as LLH values for the Glucose_state_state parameter, especially in the preprandial subcontext, are extremely useful when a physician must decide how to modify a patient's insulin regimen. Furthermore, once created, the prevalence of such patterns can be calculated – an important step in determining whether the pattern is a common one for the patient. Glucose_state_state values that are measured within *different phases* (e.g., prelunch and presupper), but within the *same day*, can be joined by interpolation within the same *generalized* (see section 4.2) interpretation context (actually, a nonconvex version of the generalized PREPRANDIAL context interval; see section 4.3) creating an abstraction comprising several preprandial abstractions, up to 6 to 8 hours apart. The maximal gap is defined by a interphase Δ function. Diurnal state abstractions that are measured in the *same phase* but over *different* (usually consecutive) *days*, such as several values of the Glucuse_state_DM_prebreakfast parameter, can be joined by interpolation within the same interpretation context (e.g., a nonconvex PREBREAKFAST context interval, that comprises all breakfasts within a given interval), up 24 to 28 hours apart, using another interphase Δ function.

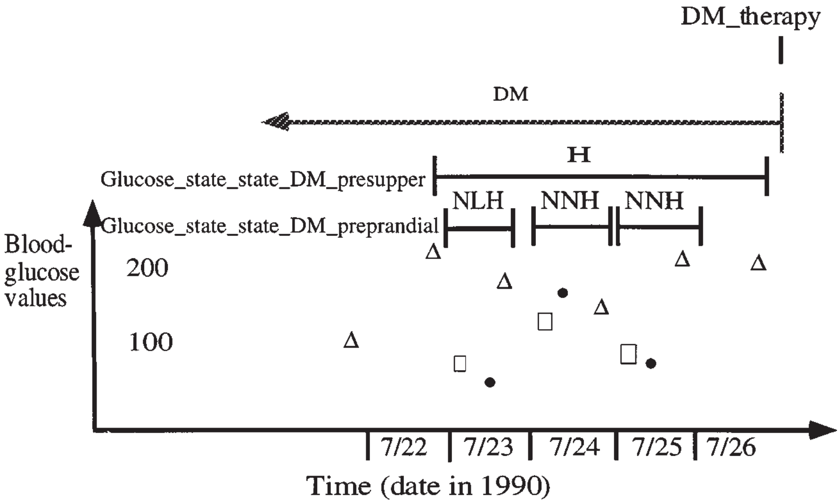


Figure 9. Abstraction of data by the RÉSUMÉ system in case 3. The DM-therapy abstraction goal induces a retrospective interpretation context, within which abstractions of blood glucose can be formed. = (open) context interval; = abstraction interval; □ = prebreakfast glucose; • = prelunch glucose; Δ = presupper glucose; DM = diabetes mellitus therapy context; GLSS_DM_PS = Glucose_state_state abstraction in the DM and presupper context; GLSS_DM_PREPRANDIAL = Glucose_state_state abstraction in the DM and preprandial context.

The two experts formed (independently) temporal abstractions from more than 800 points of data, representing two weeks of glucose and insulin data from each of eight patients. The RÉSUMÉ system created 132 (80.4%) of the 164 temporal abstractions noted by both experts [25]. Several temporal patterns were not detected, such as certain periodic patterns, partially due to the limited expressivity of the current pattern-matching language, and partially due to lack of general domain knowledge (e.g., recurrence of high blood glucose during noon time at weekends is due to the patient's lessened diet control) [25].

An example of the RÉSUMÉ output is shown in figure 9. In the particular time window shown, two significant abstractions are demonstrated:

(1) A period of 5 days of HIGH presupper blood-glucose values was created by the abstraction process. This abstraction was returned in response to a query for a period of at least 3 days of the Glucose_state_state parameter, with value HIGH, in the presupper [nonconvex] context.

(2) A set of three Glucose_state_state abstractions representing a repeating diurnal pattern, consisting of NORMAL or LOW blood-glucose levels during the morning and lunch measurements, and HIGH glucose levels during the presupper measurements. Individual abstractions in the set were created by the abstraction process; the whole set was returned in response to a query for Glucose_state_state values in the preprandial [nonconvex generalized] context (i.e., the context in which blood-glucose values, measured before several different consecutive meals, are

abstracted). The combined pattern is in fact a well-known one for experts, and suggests that an adjustment of the intermediate-acting insulin (e.g., NPH) may be indicated. This pattern was noted in the data by both experts.

Examination of the output for the first three cases by one of the experts showed that the expert agreed with almost all (97%) of the produced abstractions – a result similar to the one we found in the domain of growth monitoring [14,21]. We expected this high predictive value, since the domain's TA ontology directly reflected that expert's knowledge about these low- and intermediate-level abstractions.

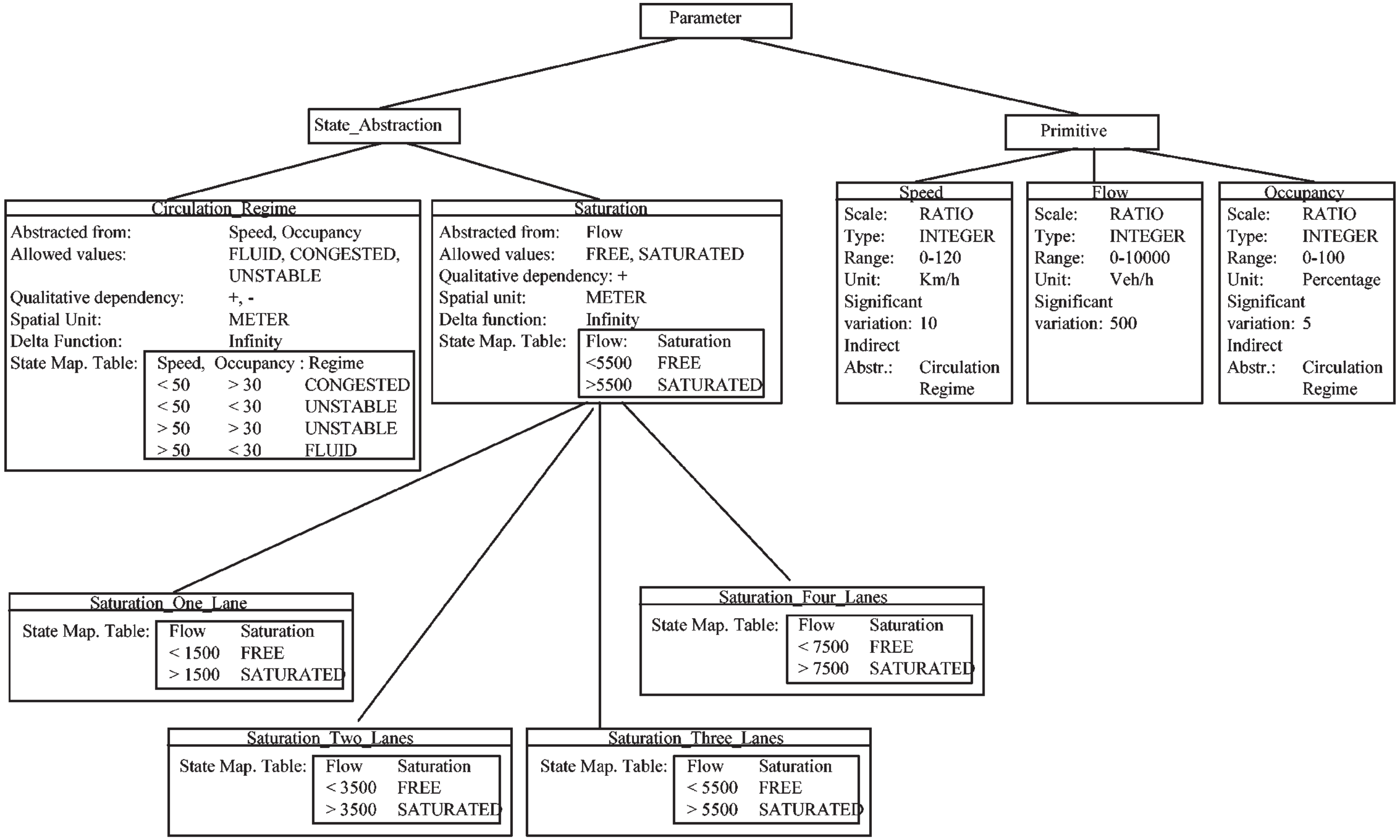
5.2. Interpretation contexts in evaluation of traffic-control actions

We have applied the RÉSUMÉ system to a task that involves both spatial and temporal abstraction: evaluation of traffic control actions [18,23]. In the traffic-control domain, the RÉSUMÉ system was used to model the task of monitoring traffic-control actions, and to create a prototype for solving that task [23]. The task of monitoring traffic-control actions receives as input recent values of different road parameters (speed, flow, and occupancy) measured by sensors located along several highways, and a set of recent control actions (e.g., traffic diversion) undertaken by traffic controllers. It returns an evaluation of the adequacy of the control actions. Performance of this task requires both *temporal* reasoning (e.g., about durations, rates, and trends of traffic parameters over time, for a given location) and linear *spatial* reasoning (e.g., about queue lengths along the highway, at a given time). We used the RÉSUMÉ system to model and solve *both* tasks [23] by generalizing it into a **knowledge-based linear-abstraction method**. This method can then be used twive, once to solve a **spatial-abstraction** (**SA**) task and once to solve a TA task. In fact, two copies of the method, using different domain-specific knowledge, can be assembled to create a **spatiotemporal-abstraction method** [18]. One of the copies can be viewed as the KBTA method, and the other as a **knowledge-based spatial-abstraction (KBSA) method**.

5.2.1. Spatial abstraction in the traffic-control domain

We defined a *spatial-abstraction ontology*, using the TA ontology knowledge structures, to describe properties of spatial parameters, such as CONGESTION, along the highway-distance dimension (figure 10). We used this ontology to create linear spatial abstractions in each highway zone, such as a spatial interval [1500, 2000] of length 500 meters in zone 1 in which the SATURATION-level parameter has the value critical. Such a spatial abstraction holds, of course, only for an instantaneous temporal snapshot.

The *primitive parameters* and respective units in the traffic domain are the three basic magnitudes recorded by sensors which are the inputs of the system: speed (*km/h*), flow (*vehicle/hr*), and occupancy (*percentage*). The *abstract parameters* are high-level qualitative variables representing the state of the traffic and several intermediate variables in the abstraction process, such as SATURATION degree (*percentage*), CIRCULATION (on of FLUID, UNSTABLE, CONGESTED), and SATURATION level (one of FREE, CRITICAL).



Spatial Ontology (partial view)

Figure 10. A part of the spatial-abstraction (SA) parameter ontology in the traffic-control domain. Note the specialization by the lane-type interpretation contexts.

The SATURATION level is abstracted from the SATURATION degree. The CIRCULATION parameter is abstracted from speed and occupancy.

Interpretation contexts in the SA task within the traffic-control domain include, for example, the number of lanes of the highway (see figure 10). Interpretation contexts are induced also by *events* such as the presence of an accident blocking one or more lanes, a road construction reducing the capacity of a freeway section, or the state of a reversible lane. Interpretation contexts determine how parameters such as the SATURATION degree should be abstracted (e.g., the SATURATION degree for one lane is $100 \times \text{flow}/1600$, but is $100 \times \text{flow}/3500$ for two lanes).

Other types of knowledge are represented in the traffic-domain's SA ontology, besides vertical-classification knowledge. One is the Δ (maximal-gap) global persistence function which expresses the maximal distance between successive disjoint parameter intervals that still allows joining them into a new parameter interval through interpolation. Thus, in the case of the CIRCULATION parameter and the CONGESTED value, this distance could be established as 3 km (i.e., two CIRCULATION-parameter intervals with the CONGESTED value would be joined into a longer interval when the distance between the endpoint locations was less than 3 km; if the distance was bigger, they would be interpreted as two different problems). This particular feature of the KBSA method is especially useful in the traffic domain since sometimes sensors do not work, certain data are missing, and the system must be able to interpolate using other sensors and heuristics.

Values for the SA knowledge type depend on particular highways. The approach we used here was to consider each highway (and even highway zone) as a different interpretation context, and specialize the SA ontology by these contexts. Using the SA ontology, we created spatial abstractions using a spatial-abstraction version of RÉSUMÉ and values from simulated highway data sensors (figure 11).

5.2.2. Temporal abstraction in the traffic-control domain

We created a TA ontology (figure 12) to describe the properties of the temporal evolution of the spatial abstractions at each location or highway zone (each of which is an instantaneous snapshot) over time. We used the TA ontology to form and detect crucial traffic-control spatiotemporal patterns. In this case, the primitive parameters include values provided by the output of the SA task and values provided by sensors at critical points outside the highway, such as ramps or intersections: CONGESTION length (*meters*), FLOW at point P_i (*vehicles/hr*) (the number i of these points is usually less than 5 per highway).

For the sake of clarity, we assume that a highway can have at most one problem at a time. In fact this is normally true. However, reasoning about multiple problems is not difficult; several *zones*, as they are often called, must be defined, with each zone corresponding to a spatial interval between two consecutive message-sign devices. Zones can be represented as *subcontexts* (a part of the TA and SA interpretation-context ontologies). A traffic queue usually has a fixed starting point where there is a lack of capacity (an accident, a bottleneck, etc.) and the end of the congested area

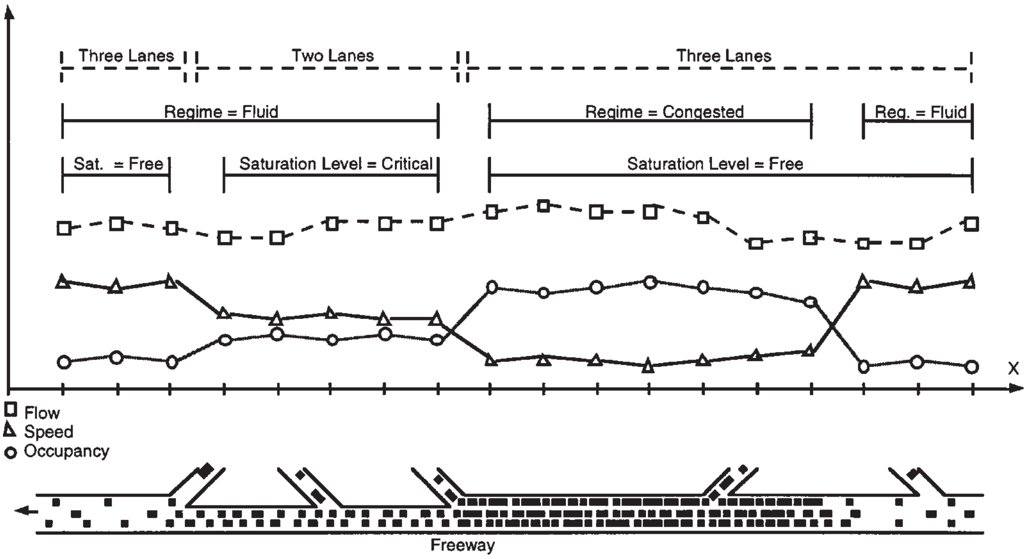


Figure 11. Spatial abstraction of a highway section. At the bottom, a scheme of the highway is presented showing different densities of traffic. Above that, respective values are presented for different magnitudes recorded by sensors at consecutive locations (SPEED, OCCUPANCY and FLOW). At the top, different intervals are shown as a result of the spatial abstraction. The figure shows only top-level abstraction intervals although different intermediate intervals are also created during the inference process. □ = FLOW; Δ = SPEED; ○ = OCCUPANCY; ⊢ — ⊣ = context interval; ⊢—⊣ = closed abstraction interval.

evolves according to the demand. This means that if there were several problems on the same highway, each one could be identified by the zone of their starting point. Figure 13 presents an example of a temporal evolution of the CONGESTION-length parameter in one zone.

Temporal interpretation contexts in the traffic-control domain are typically induced by events (execution of traffic-control actions), such as the action of turning on a congestion warning at a certain zone, or the creation of a path diversion.

The abstract parameters of the TA ontology for the control-monitoring task in the traffic domain include: CONGESTION-length gradient (one of INCREASING, DECREASING, CONSTANT); FLOW gradient at point P_i (one of INCREASING, DECREASING, CONSTANT); SATURATION level at point P_i (one of FREE, CRITICAL).

The CONGESTION-length gradient is necessary to decide if the control action is having an effect on the existing problem. Flow gradients monitor whether control actions such as diversion are followed by drivers. The SATURATION level at critical points is useful to determine whether a new problem may appear as a consequence of the control action. Vertical-classification tables for the SATURATION level are specialized by each subcontext created by each point P_i. The horizontal-inference knowledge for gradient interpolation includes values of variations significant to the values of the parameters abstracted (e.g., 1000 m for CONGESTION length, 500 vehicles/hr for the FLOW parameter).

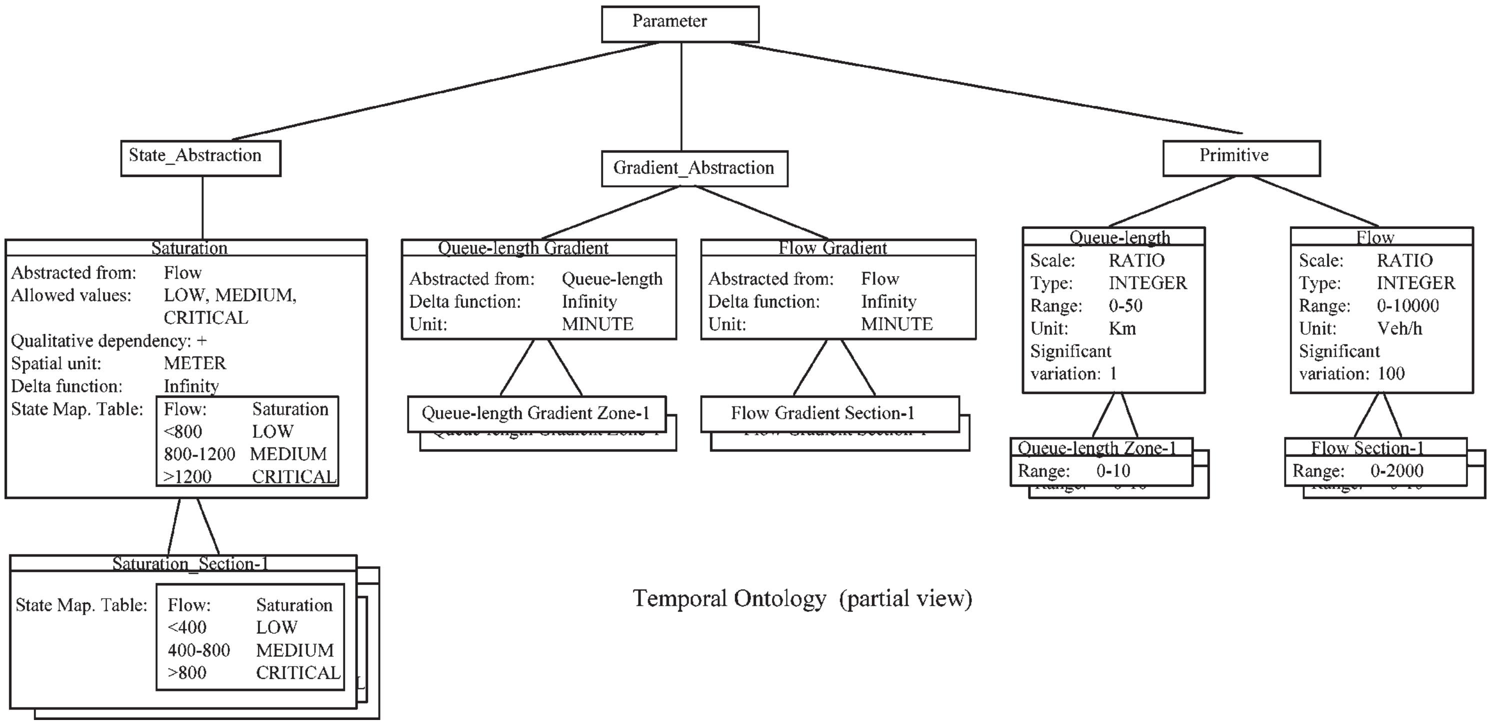


Figure 12. A part of the temporal-abstraction parameter ontology in the traffic-control domain. Note the specialization by geographic-region interpretation contexts.

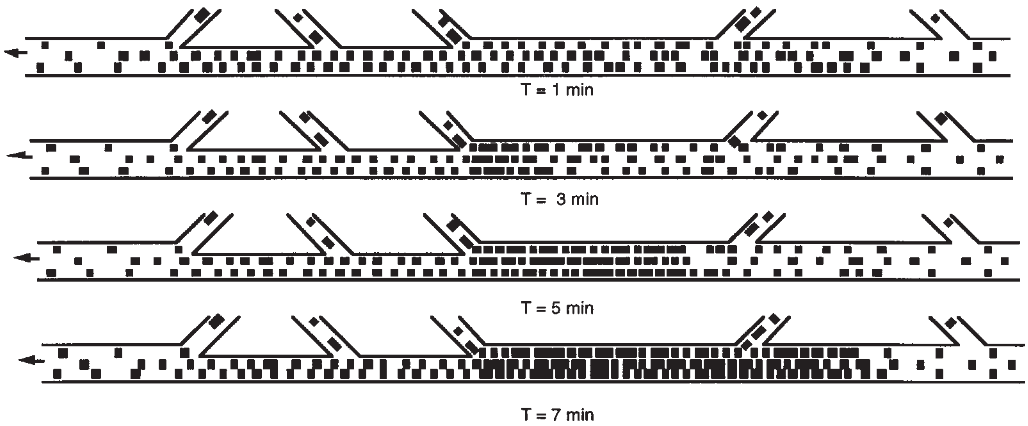


Figure 13. The input for the temporal abstraction task in the domain of traffic is a sequence of qualitative instantaneous views of the same segment of the highway. The figure presents the temporal evolution of the state of a highway in which the congestion-length gradient in a certain zone can be abstracted as increasing over successive temporal snapshots (at T = 1 min, T = 3 min, T = 5 min, and T = 7 min).

Finally, to determine the adequacy of a control action it is necessary to define *temporal patterns*. For each control action we defined the following set of TA patterns representing its adequacy: APPROPRIATE, USELESS, NEGATIVE, UNKNOWN, SOLVED and INSUFFICIENT. Each pattern is expressed as a set of parameter intervals and temporal and value constraints among them. The values of these patterns (one of TRUE, FALSE) supplied the final answer to the control-action monitoring task. Figure 14 shows an example of a temporal abstraction of the spatial data abstracted from one highway section, showing an evolution of its (abstract) parameters over time.

6. Related work

Several general frameworks have been proposed for explicitly representing the context of inference [3–5,9,16] and for enabling reasoners to *lift* axioms from one context to another [9]. Our framework is more task specific, focusing on the TA task and on contexts as being interpreted only over time intervals, and can be viewed as an engineering approach to the representation of contexts over time.

DIRCs (and, in a certain sense, the various types of interpretation contexts) represent explicitly the *source* of the contexts (i.e., induction by context-forming propositions and formation by certain operations on context intervals), a point often neglected in logical frameworks that assume the existence of contexts as primitive propositions.

Parameter intervals represent explicitly the parameter's value within a specific context and time. These propositions could also be represented as a parameter value that holds within a temporal specialization of the context [16]. Primitive-parameter propositions, such as values of sensor readings at a particular location and time, which are otherwise independent of context (and are thus considered to be in a null, or empty context) can be compared to time-oriented propositions that hold in the *outermost*

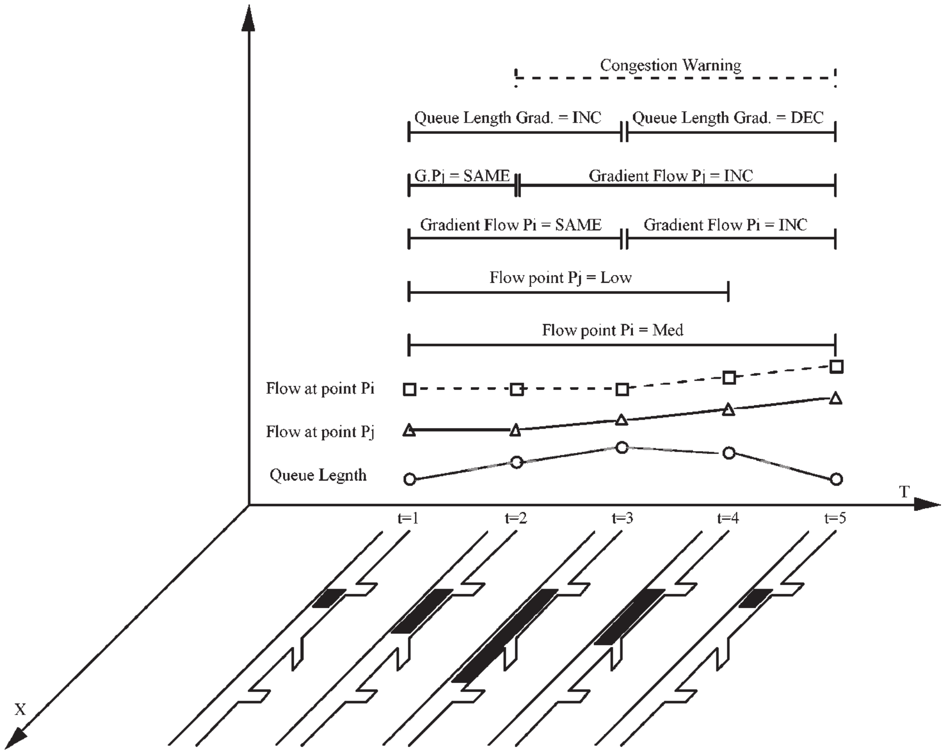


Figure 14. Temporal abstraction of a highway section. At the bottom is the evolution of the highway state over time; a queue first increases and then decreases. Above that, values are presented for the queue length and for values measured by sensors at critical spatial locations. At the top are inferred temporal abstraction intervals. ⊢ — ⊣ = context interval; ⊢—⊣ = closed abstraction interval.

context, or context C_0 [16], which can be dropped to decontextualize raw data. Alternatively, the parameter's value (as an independent proposition) could be considered to be holding in the outermost context, specialized only by time.

Both the IS-A and the SUBCONTEXT relations in the TA context ontology are *specializing* relations in McCarthy's [16] sense. However, unlike in McCarthy's case, inheritance typically works only from the super context to the subcontext or the descendent context, since typically the TA properties of the parameter in the more specialized context are somewhat different and override the properties of that parameter in the more general context (else, the domain expert would not have defined that specialization). The IS-A relation specializes contexts in the regular sense. The SUBCONTEXT relation creates composite contexts, or *nested* contexts, as in McCarthy's [16] construction, thus bestowing on contexts what Buvac et al. [4,5] refer to as the *non-flatness* property. That is, the path by which the same context is arrived at is important; in our case, it is the context chain in a composite context.

Generalized contexts, as we define them (see section 4.2), enable sharing specific parameter propositions (i.e., specific values) and their meaning across different contexts, which can be considered as a special relation, or lifting rule, among contexts [9,16]. Nonconvex contexts are more akin to nonconvex temporal intervals, for which Ladkin [15] has defined a detailed taxonomy and algebra.

Somewhat related to induction of contexts by propositions is Kowalski and Sergot's [13] *event calculus*, in which *states* are initiated by *events*. That formalism, however, is intended mainly for a *forecasting* or a *projection* task, and less for an *interpretation* task, such as the TA task. Thus, it is well suited for projection of future results of various operations in a database, for instance, and for simulation or prediction of likely immediate results of actions, tasks for which the KBTA framework is not intended and therefore cannot perform. However, the KBTA framework, due to the specialized task it performs, represents explicitly the interpretation contexts created by either past or future events and states; furthermore, any quantitative temporal constraint can be defined to hold between the inducing and the induced proposition. Both past and delayed effects are outside of the scope of formalisms such as the event calculus, in which events directly and instantaneously create (future) state transitions. Note that DIRCs do not represent effects, but rather, the context(s) induced by either the presence of the proposition, the potential effects of the proposition, or the abductive inferences from realizing the existence of the proposition. Thus, there is no intent to simulate a real sequence of actions and effects.

In the medical domain, several systems have been applied to the abstraction of meaningful time-oriented patterns from clinical data. Several of these systems represent the context explicitly or implicitly, and thus can be compared to the KBTA method and to the RÉSUMÉ system.

Fagan's **VM** system was one of the first knowledge-based systems that included an explicit representation for time. It was designed to assist physicians who are managing patients who were on ventilators in intensive-care units [7,8]. VM was designed as a rule-based system inspired by MYCIN [2], but it was different in several respects: VM could reason explicitly about time units, accept time-stamped measurements of patient parameters, and calculate time-dependent concepts such as rates of change. In addition, VM relied on a state-transition model of different intensive-care therapeutic situations, or *contexts* (in the VM case, different ventilation modes). In each context, different *expectation rules* would apply to determine what, for instance, is an ACCEPTABLE mean arterial pressure in a particular context. Except for such state-specific rules, the rest of the rules could ignore the context in which they were applied, since the context-specific classification rules created a context-free, "common denominator," symbolic-value environment. Thus, similar values of the same parameter that appeared in meeting intervals (e.g., IDEAL mean arterial pressure) could be joined and aggregated into longer intervals, even though the meaning of the value could be different, depending on the context in which the symbolic value was determined. The fact that the system changed state was inferred by special rules, since VM was not connected directly to the ventilator output.

The role of the context-forming mechanism in RÉSUMÉ (namely, to create correct interpretation contexts for temporal abstraction) is not unlike that of the state-detection rules in VM, although the mechanism's operation is different and its output is more flexible (e.g., the temporal extension of an interpretation context can have any of Allen's 13 temporal relations to the event or abstraction which induced it, not just a contemporaneous scope).

The RÉSUMÉ methodology is similar in some respects to the VM model. Most of the domain-specific knowledge that is represented in the domain's ontology of parameters and their temporal properties is specialized by contexts. This knowledge is used by the temporal-abstraction mechanisms. Thus, although, strictly speaking, the same domain-independent rules (the TA mechanisms) apply to every context, their parameters (e.g., classification tables, maximal-gap-bridging functions) are specific to the context. However, the various values and classification functions possible for the same parameter or a combination of parameters in each context can be quite different. Thus, the GRADE_ IV value of the SYSTEMIC_TOXICITY abstract parameter (and, often, the whole parameter schema itself) makes no sense when a patient received no toxic chemotherapy, even though the same hematological parameters might still be monitored. This context-specific terminology is termed the *context's vocabulary* by Buvac and Mason [5]. Therefore, additional conditions must be specified before meeting parameter intervals with the same parameter value can be joined. As explained in sections 4.2 and 4.3, the KBTA method makes several finer distinctions with respect to joining parameter values over different contexts: typically, an interpretation of the same parameter in different contexts cannot be joined to an interpretation of that parameter in different contexts. However, the RÉSUMÉ system allows the developer to define *generalizing, interpretation contexts* for joining interpretations of the same parameter in different contexts over meeting time intervals (e.g., the state of the hemoglobin parameter over two meeting but different treatment regimens within the same clinical protocol), and *nonconvex interpretation contexts* for joining interpretations of the same parameter in the same context, but over nonconsecutive time intervals (e.g., prebreakfast blood-glucose values over several days). Note that, in the terms of the TA ontology defined in section 4.2, all of VM's parameter propositions are *sharable*, for all parameter values. This phenomenon was enabled by the fact that all contexts in VM had the same vocabulary, in particular with respect to the parameters' domain of values and its meaning.

The **TrenDx** system of Haimowitz and Kohane [10] builds on Kohane's [12] constraint-satisfaction *temporal-utilities package*, and defines domain-specific patterns called *trend templates* (*TTs*). TrenDx is useful in detecting that the data is consistent with one or more TTs, including TTs of which only a part is observed. Like RÉSUMÉ, TrenDx assumes implicitly an ill-defined domain that cannot be modeled easily quantitatively, and therefore requires detection of essentially associative temporal patterns. However, the goal of TrenDx is different from that of RÉSUMÉ. TrenDx does not create any intermediate abstractions or context intervals, since its goal is not to abstract, summarize, or answer new queries about the data, as it is in the TA task, but rather to match data efficiently against a set of predefined patterns. Data can only be accepted at the lowest level; thus, no input of intermediate-level abstractions is possible. Contexts are therefore an implicit part of the pattern, and no metalevel knowledge exists about them. No explicit domain ontology of parameters, events, and contexts exists, and a constraint (e.g., significant change in a parameter) might be repeated with the same implicit role in different TTs (when used within what might be referred to as the same [implicit] context) and even in different parts of the same TT.

Kahn's **TOPAZ** system [11] integrates a quantitative physiological model and a symbolic model for aggregation of clinically significant intervals. TOPAZ can associate interpretation methods with an interval representing a *context* of interest. RÉSUMÉ extends this capability by the context-forming mechanism, which uses an explicit context ontology to enable creation of context-specific abstractions and activation of specific functions, but does not limit generated interpretation contexts to the temporal extent of the parent event, allowing any desired relation between the generating interval and the generated context.

7. Discussion: advantages and limitations

Interpretation contexts enable temporal abstractions to be sensitive to the most specific context known at the time of interpretation, while limiting the computation by focusing the attention of the TA mechanisms on a small portion of the input (i.e., to parameter propositions within the temporal scope of a context interval).

As explained in section 6, the KBTA method is effective mainly for *interpretation* tasks, but not for *forecasting*, *simulation*, or *projection* tasks. In addition, the explicit context ontology must be acquired from domain experts and has to be defined ahead of time (although context intervals are induced or extended automatically at run time). Thus, the KBTA method implies a certain *knowledge-acquisition* (*KA*) cost, and all relevant interpretation-context combinations used in the parameter ontology have to be valid composite contexts that are defined (explicitly or implicitly) in the context ontology. To facilitate KA and validation, we have created an automated graphic KA tool for acquisition of TA knowledge [29]. The TA KA tool is generated automatically by tools from the PROTÉGÉ project for automatic generation of KA tools for knowledge-based systems [30].

The TA mechanisms (except for context formation) operate within the temporal span of context intervals and *do not depend on the event and context ontologies*. These mechanisms assume the existence of context intervals and of interpretation contexts as part of the input parameter propositions. The context-forming mechanism is thus the only interface to the domain's event and abstraction-goal ontologies, and shields the rest of the mechanisms from any need to know about external events, their structure, or the interpretation contexts they induce.

The introduction of explicit interpretation contexts as independent mediating entities, separate from the propositions inducing them and from abstractions using them, has significant conceptual and computational advantages for context-specific interpretation of time-stamped data. Advantages include:

(1) Any temporal relation can hold between a context interval and its inducing proposition; interpretation contexts might be induced concurrently, in the future, and in the past, enabling a form of foresight and hindsight.

(2) The *same context-forming proposition* can *induce* one or more context intervals.

(3) The *same interpretation context* might be induced by *different* propositions. The separation of interpretation contexts from their inducing propositions facilitates maintenance and reusability of temporal-abstraction knowledge.

(4) Parameter propositions include an explicit interpretation context, thus enabling a representation of several abstractions in which the *same abstract parameter* (e.g., the "state of hemoglobin-level" parameter) has *different values* at the *same time* – one for each of the context intervals that hold during the relevant period. Thus, interpretation contexts support maintenance of several *concurrent* interpretations of the same data.

Apart from its demonstrated usefulness in several clinical domains, such as the diabetes-therapy domain example presented in section 5.1, the interpretation-context model has been useful also in modeling other tasks involving matching of context-sensitive linear patterns in which the relevant distance measure has properties similar to the temporal model defined in the TA ontology (see section 3). One example is modeling the retrieval of full-text documents, given key words that should be found within a given semantic text context [26]. In that task, the linear distance measure was position within the text, the parameters were text strings, and the interpretation contexts were conceptual textual contexts (e.g., a POPULATION-SELECTION subcontext within a clinical research paper). Another example, discussed in section 5.2, was assistance in the formation of both temporal and spatial abstractions to solve a traffic-control task. The linear distance measure was either time or space, respectively, and the KBTA method and the RÉSUMÉ system were used for both tasks [23]. Indeed, a *spatio-temporal* abstraction method was assembled by reusing two copies of a generalized knowledge-based linear-abstraction method [18] and mapping the inputs and outputs of that method to either a spatial-abstraction or a temporal-abstraction domain ontology, as necessary.

Thus, the interpretation-context model presented in this paper might also be viewed as a general model for knowledge-based creation of interpretation contexts for a linear-abstraction task. The necessary conditions include an distance measure analog to the temporal one defined in the TA ontology. To make the knowledge-based linear-abstraction method not only applicable but also useful, however, there are also several additional requirements for successful reuse [18]. For instance, certain functional specifications of the task make this method more valuable (e.g., parameters exist at several levels of abstraction, and data values might arrive out of the linear order) and certain domain-specific TA knowledge or its equivalent must be available (e.g., context-forming classification knowledge and ABSTRACTED-INTO relations over the parameter space).

Acknowledgements

This work has been supported by grants LM05708 and LM06245 from the National Library of Medicine, and IRI-9528444 from the National Science Foundation. Computing resources were provided by the Stanford CAMIS project, funded under Grant No. LM05305 from the National Library of Medicine. I thank Mark Musen, Richard Fikes, and Barbara Hayes-Roth for advice and support. I had useful discussions with Samson Tu, Amar Das, and Michael Kahn.

References

[1] J.F. Allen, Towards a general theory of action and time, Artificial Intelligence 23(2) (1984) 123–154.
[2] B.G. Buchanan and E.H. Shortliffe, eds., *Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project* (Addison-Wesley, Reading, MA, 1984).
[3] S. Buvac, Qunatifactional Logic of Context, in: *Proc. Thirteenth National Conf. Artificial Intelligence* (AAAI Press, Menlo Park, 1996) pp. 600–606.
[4] S. Buvac, V. Buvac and I.A. Mason, Metamathematics of contexts, Fundamenta Informaticae 23(3) (1995).
[5] S. Buvac and I.A. Mason, Propositional logic of context, in: *Proc. Eleventh National Conf. Artificial Intelligence* (1993).
[6] H. Eriksson, Y. Shahar, S.W. Tu, A.R. Puerta and M.A. Musen, Task modeling with reusable problem-solving methods, Artificial Intelligence 79(2) (1995) 293–326.
[7] L.M. Fagan, VM: Representing time dependent relations in a medical setting, Ph.D. dissertation, Department of Computer Science, Stanford University, Stanford, CA (1980).
[8] L.M. Fagan, J.C. Kunz, E.A. Feigenbaum and J.J. Osborn, Extensions to the rule-based formalism for a monitoring task, in: *Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, eds. B.G. Buchanan and E.H. Shortliffe (Addison-Wesley, Reading, MA, 1984) pp. 397–423.
[9] R.V. Guha, Contexts: A formalization and some applications, Ph.D. dissertation, Department of Computer Science, Report No. STAN-CS-91-1399, Stanford University, Stanford, CA (1991).
[10] I.J. Haimowitz and I.S. Kohane, Automated trend detection with alternate temporal hypotheses, in: *Proc. Thirteenth Internat. Joint Conf. Artificial Intelligence* (Morgan Kaufmann, San Mateo, 1993) pp. 146–151.
[11] M.G. Kahn, Combining physiologic models and symbolic methods to interpret time-varying patient data, Methods of Information in Medicine 30(3) (1991) 167–178.
[12] I.S. Kohane, Temporal reasoning in medical expert systems, Technical Report 389, Laboratory of Computer Science, Massachusetts Institute of Technology, Cambridge, MA (1987).
[13] R.A. Kowalski and M.J. Sergot, A logic-based calculus of events, New Generation Computing 4 (1986) 67–95.
[14] M.M. Kuilboer, Y. Shahar, D.M. Wilson and M.A. Musen, Knowledge reuse: Temporal-abstraction mechanisms for the assessment of children's growth, in: *Proc. Seventeenth Ann. Symp. Computer Applications in Medicine*, Washington, DC (1993) pp. 449–453.
[15] P. Ladkin, Time representation: A taxonomy of interval relations, in: *Proc. Sixth National Conf. Artificial Intelligence*, Philadelphia, PA (1986) pp. 360–366.
[16] J. McCarthy, Notes on formalizing context, in: *Proc. Thirteenth Internat. Joint Conf. Artificial Intelligence* (Morgan Kaufmann, San Mateo, 1993) pp. 555–560.
[17] J. McCarthy and P. Hayes, Some philosophical problems from the standpoint of artificial intelligence, in: *Machine Intelligence* (Edinburgh University Press, Edinburgh, UK, 1969).

[18] M. Molina and Y. Shahar, Problem-solving method reuse and assembly: From clinical monitoring to traffic control, in: *Proc. Tenth Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Alberta, Canada, Vol. 1 (1996) pp. 19-1–19-20.

[19] 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 Association 3(6) (1996) 367–388.

[20] T.A. Russ, Using hindsight in medical decision making, in: *Proc. Thirteenth Ann. Symp. Computer Applications in Medical Care*, ed. L.C. Kingsland (IEEE Computing Society Press, Washington, 1989) pp. 38–44.

[21] Y. Shahar, A knowledge-based method for temporal abstraction of clinical data, Ph.D. dissertation, Program in Medical Information Sciences, Knowledge Systems Laboratory Report No. KSL-94-64, Department of Computer Science Report No. STAN-CS-TR-94-1529, Stanford University, Stanford, CA (1994).

[22] Y. Shahar, A framework for knowledge-based temporal abstraction, Artificial Intelligence 90(1–2) (1997) 79–133.

[23] Y. Shahar and M. Molina, Knowledge-based spatiotemporal abstraction, in: *Notes of the AAAI-96 Workshop on Spatial and Temporal Reasoning*, Portland, OR (1996) pp. 21–29.

[24] Y. Shahar and M.A. Musen, RÉSUMÉ: A temporal-abstraction system for patient monitoring, Computers and Biomedical Research 26(3) (1993) 255–273. Reprinted in: *Yearbook of Medical Informatics 1994*, eds. J.H. van Bemmel, T. McRay and F.K. Schattauer (The International Medical Informatics Association, Stuttgart, 1994) pp. 443–461.

[25] Y. Shahar and M.A. Musen, Knowledge-based temporal abstraction in clinical domains, Artificial Intelligence in Medicine 8(3) (1996) 267–298.

[26] Y. Shahar and G. Purcell, The context-sensitive pattern-matching task, in: *Working Notes of the Workshop on Modelling Context in Knowledge Representation and Reasoning*, Internat. Joint Conf. Artificial Intelligence, Montreal, Québec, Canada (1995) pp. 133–143.

[27] Y. Shahar, S.W. Tu and M.A. Musen, Knowledge acquisition for temporal-abstraction mechanisms, Knowledge Acquisition 4(2) (1992) 217–236.

[28] Y. Shoham, Temporal logics in AI: Semantical and ontological considerations, Artificial Intelligence 33(1) (1987) 89–104.

[29] A. Stein, M.A. Musen and Y. Shahar, A knowledge-acquisition tool for temporal abstraction, in: *Proc. 1996 AMIA Ann. Fall Symp.* (formerly Symp. Computer Applications in Medical Care), Hanley and Belfus, PH (1996) pp. 204–208.

[30] S.W. Tu, H. Eriksson, J.G. Gennari, Y. Shahar and M.A. 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) (1995) 257–289.