

Querying probabilistic temporal constraints for guideline interaction analysis: GLARE's approach

Antonella Andolina¹, Luca Anselma², Luca Piovesan³ and Paolo Terenziani³

¹ ITCS Sommeiller, Corso Duca degli Abruzzi 20, 10129 Torino, Italy

² Dipartimento di Informatica, Università di Torino, Corso Svizzera 185, 10139 Torino, Italy

³ DISIT, Università del Piemonte Orientale "A. Avogadro", Alessandria, Italy
antoando@libero.it, anselma@di.unito.it, luca.piovesan@uniupo.it
paolo.terenziani@uniupo.it

Abstract. The treatment of patients affected by multiple diseases (*comorbid* patients) is one of the main challenges of the modern healthcare, involving the analysis of the interactions of the guidelines for the specific diseases. However, practically speaking, such interactions occur *over time*. The GLARE project explicitly provides knowledge representation, temporal representation and temporal reasoning methodologies to cope with such a fundamental issue. In this paper, we propose a further improvement, to take into account that, often, the effects of actions have a *probabilistic distribution in time*, and being able to *reason* (through constraint propagation) and to *query probabilistic temporal constraints* further enhances the support for interaction detection.

Keywords: probabilistic temporal constraints, temporal reasoning, guideline interaction analysis, decision support system

1 Introduction

Clinical practice guidelines are the major tool that has been introduced to grant both the quality and the standardization of healthcare services, on the basis of *evidence-based* recommendations. The adoption of computerized approaches to acquire, represent, execute and reason with Computer-Interpretable Guidelines (CIGs) provides crucial additional advantages so that, in the last twenty years, many different approaches and projects have been developed to manage CIGs (consider, e.g., the book [1] and the recent survey [2]). One of such approaches is GLARE (Guideline Acquisition, Representation and Execution) [3], and its successor METAGLARE [4]. By definition, clinical guidelines address specific pathologies. However, *comorbid* patients are affected by more than one pathology. The problem is that, in comorbid patients, the treatments of single pathologies may interact with each other, and the approach of proposing an ad-hoc "combined" treatment to cope with each possible comorbidity does not scale up. In the last years, several computer-based approaches

have started to face this problem and also GLARE has been extended to cope with comorbid patients. In this paper we focus on interaction detection. In [5] we developed an ontology for interactions, and complemented it with detection algorithms. Interactions between CIGs occurs *over time*. Indeed, the effects of two actions taken from different guidelines can practically conflict only if the times of execution of such actions are such that their effects overlap in time. In [6] we proposed an explicit treatment of temporal constraints and of temporal reasoning in GLARE. However, such previous approaches disregard the fact that temporal constraints may have different *probabilities*, and such probabilities may be important for physicians to correctly analyze and manage interactions.

Our running example considers drug interactions. Several aspects influence the absorption of a drug, and therefore its effects. In particular, they are influenced by the methods of administration (e.g., enteral, parenteral, transcutaneous...) of the drug, by its mechanisms of absorption and elimination, and by the targets of the administered substance. The fields in medicine that study such mechanisms are the pharmacokinetics and pharmacodynamics. It integrates a pharmacokinetic and a pharmacodynamic model component into one set of mathematical expressions that allows the description of the time course of effect intensity in response to administration of a drug dose. Deriving from such mathematical expressions the probabilities of the effects of a drug along time is difficult. As an approximation, we have considered the models of the plasma concentrations of the drugs, their half-life (i.e., the time to reduce the substance amount in the blood of 50%) and the type of the effect, and we approximate the probabilities with the help of an expert.

Example 1. Consider, for instance, a patient affected by gastroesophageal reflux (GR) and by urinary tract infection (UTI). The CIG for GR may recommend calcium carbonate administration (CCA; assumed to be punctual at the chosen temporal granularity), to be administered within three hours. CCA has the effect of decreasing gastric absorption (DGA). Considering as *granularity units of 15 minutes*, DGA can start after 1 unit with probability 0.4, after 2 with probability 0.4, and after 3, with probability 0.2. Additionally, the duration of DGA may be 4 units (probability 0.1), 5 (0.3), 6 (0.4), 7 (0.1), or 8 (0.1). The CIG for UTI may recommend Nalidixic acid administration (NAA), to be administered within two hours. NAA has as effect Nalidixic acid gastric absorption (NAGA), starting after 1 unit (probability 0.4) or 2 (probability 0.6). The duration of NAGA may be 1 (probability 0.05), 2 (0.05), 3 (0.15), 4 (0.15), 5 (0.25), 6 (0.25), 7 (0.05), 8 (0.05). ■

In order to support physicians in the study of the interaction between CCA and NAA, one must take into account not only the temporal constraints, but also their probabilities. This is essential in order to answer physician's queries such as:

(Q1) *If I perform on the patient CCA in unit 1 or 2 (i.e., in the following 30 minutes), and NAA in units 1 or 2 (i.e., in the following 30 minutes), what is the probability that the effects of such two actions intersect in time (i.e., what is the probability of the interaction between CCA and NAA)?*

In the following, we sketch our ongoing approach to support physicians in the management of probabilistic temporal interaction detection. This is the first approach in the literature managing such a challenging task. Specifically, to the best of our knowledge, our approach is the first one that:

3 Querying probabilistic temporal constraints for guideline interactions analysis in GLARE

- (1) Introduces *probabilistic quantitative temporal constraints*, and provide a *constraint propagation algorithm* to reason with them
- (2) Identify a comprehensive *query language* operating on such constraints
- (3) Provides a support to *evaluate the queries*
- (4) Proposes the introduction of such mechanisms (which are domain independent) in the analysis of *temporal interactions* between guidelines.

Notably, while contribution (1) already appeared in a recent work [7], results (2—4) are entirely new contributions of this paper.

2 Background and related work

Temporal constraints and temporal reasoning. Informally speaking, temporal constraints are limitations of the possible time of occurrence of events. *Quantitative* temporal constraints involve *metric time* and are very frequent in many application domains. They include *dates* (e.g., “John arrived on 10/10/99 at 10:00”), *durations* (e.g., “John worked for 3 hours”) and *delays* (e.g., “John arrived 10 minutes after Mary”). *Qualitative* temporal constraints concern the relative position of events (e.g., “John arrived at work after Mary (arrived)”). Notably, in many cases, temporal constraints are *not exact* (e.g., “John arrived between 10 and 30 minutes after Mary”). A plethora of approaches has been developed within the AI community to deal with quantitative temporal constraints (see, e.g., the survey in [8]). However, all of them agree that, given a set of temporal constraints, *temporal reasoning* is fundamental for different tasks, including to check their *consistency*, to find a *scenario* (i.e., a solution: an instantiation of all events such that all constraints are satisfied), to make explicit the *tightest* implied constraints, and/or to *answer queries* about the (explicit plus implied) constraints. Notably, while in several task (e.g., in scheduling) the goal is to find a *scenario*, in others, such as *decision support* (which is the context of our work), the *minimal network* (representing the tightest temporal constraints) must be determined, to provide users with a compact representation of all the possible solutions (since the choice of a specific solution must be left to the users).

A well-known and widely used framework to cope with quantitative temporal constraints is STP (Simple Temporal Problem [9]). In STP, constraints have the form $P[l,u]P_i$, where P_i and P_j denote time points, and l and u ($l \leq u$) are integer numbers, stating that the *temporal distance between P_i and P_j ranges between l and u* . In most AI approaches, temporal reasoning is based on two operations on temporal constraints: *intersection* and *composition*. Given two constraints C_1 and C_2 between two temporal entities A and B , temporal intersection (henceforth \cap) determines the most constraining relation between A and B (e.g., $A[20,40]B \cap A[30,50]B \rightarrow A[30,40]B$). On the other hand, given a constraint C_1 between A and B and a constraint C_2 between B and C , composition ($@$) gives the resulting constraint between A and C (e.g., $A[20,40]B @ B[10,20]B \rightarrow A[30,60]C$).

In STP, constraint propagation can be performed taking advantage of the possibility of representing temporal constraints as a graph, and applying Floyd-Warshall’s *all-pairs shortest path* algorithm (see below, where $\lambda(i,j)$ denotes the constraint between two time points i and j).

```

for k:=1 to n do
  for i:=1 to n do
    for j:=1 to n do
       $\lambda(i, j) = \lambda(i, j) \cap (\lambda(i, k) @ \lambda(k, j))$ 

```

As discussed in [9], Floyd-Warshall’s algorithm is *correct* and *complete* on STP, operates in cubic time, and provides as output the *minimal network* of the input constraints, i.e., the tightest equivalent STP, or an inconsistency (in case a negative cycle is detected).

In the last two decades, many approaches have realized that, in many domains, “crisp” temporal constraints are not enough, since *preferences* or *probabilities* have to be considered. An important mainstream of research in this area (in which our approach is located) has focused specifically on the representation of “*non-crisp*” *temporal constraints*, and on the *propagation* of such constraints. Concerning *qualitative* constraints, in their seminal work Badaloni and Giacomini [10] have defined a new formalism in which the “crisp” qualitative temporal relations in Allen’s Interval Algebra are associated with a degree of *plausibility*, and have proposed temporal reasoning algorithms to propagate such constraints. Ryabov et al. [11] attach a *probability* to each of Allen’s basic interval relations. A similar probabilistic approach has been proposed more recently by Mouhoub and Liu [12], as an adaptation of the general probabilistic CSP framework. “*Non-crisp*” *quantitative temporal constraints* have been considered by Khatib et al. [13], that extended the STP and the TCSP framework [9] to consider temporal *preferences*. An analogous approach has been recently proposed in [14]. However, until now, no approach has been developed to cope with both quantitative temporal constraints and probabilities, and to perform *query answering* on them.

CIG interaction detection. In short, our approach is *the only approach in the CIG literature focusing on the temporal detection of CIG interactions*. Indeed, most of the CIG approaches to comorbidities do not even focus on interaction detection: they simply assume that the possible interactions are identified a-priori by physicians, and focus on how to merge the CIGs in such a way that the interactions are avoided or managed. As a remarkable exception, [15] exploits ontological knowledge and domain-independent general rules to support the *automatic detection of interactions* between (the effects of) medical actions. However, in [15] *no temporal analysis* is performed to check whether such interactions can effectively occur during the treatment of a specific patient. In our GLARE approach, a similar methodology has been devised, extending it with the possibility of *performing the temporal analysis of interactions* [6], and a methodology to support physicians in their management [16]. However, our temporal approach in [6] only considers “crisp” temporal constraints, so that the approach can only warn physicians whether an interaction certainly occurs, possibly occurs, or cannot occur, while physicians in several cases would prefer a “finer” support, considering also the probability of such occurrences. This is the task of the work in this paper.

3 Representing and reasoning with probabilistic temporal constraints

In [7] we proposed an extension of *quantitative* (i.e., metric) temporal constraints of STP [9] to support the possibility to associate preferences between alternative constraints in the form of probabilities. The distances between two points (denoting the starting/ending points of events) are a *convex* and *discrete* set of alternatives, from a minimum to a maximum distance. A probability is associated with each distance.

Definition 1. Probabilistic Quantitative Temporal Constraint (PQTC). Let $t_i, t_j \in \mathbb{Z}$ be time points. A PQTC between t_i and t_j is a constraint of the form $t_i \langle (d, p), \dots, (d, p) \rangle t_j$, where (i) $p_1, \dots, p_n \in \mathfrak{R}$ are probabilities ($0 \leq p_1 \leq 1, \dots, 0 \leq p_n \leq 1$), (ii) $d_1, \dots, d_n \in \mathbb{Z}$ are distances, and (iii) $\sum p_1, \dots, p_n = 1$. ■

The intended *meaning* of a constraint $t_i \langle (d, p), \dots, (d, p) \rangle t_j$ is that the *distance* $t_i - t_j$ between t_i and t_j is d with probability p , or ... or d with probability p .

Note. In PQTCs, we assume that the distances d_1, \dots, d_n are ordered.

A PQTC $t_i \langle (d, p), \dots, (d, p) \rangle t_j$ can be graphically represented by an directed arc labelled $\langle (d, p), \dots, (d, p) \rangle$ connecting two nodes N_i and N_j , representing the time points t_i and t_j respectively.

Definition 2. Probabilistic Temporal Network (PTN). Given a set $V = \{t_1, \dots, t_n\}$ of time points, a Probabilistic Temporal Network (over V) is a set of probabilistic quantitative temporal constraints over V . It can be graphically represented by a directed graph. ■

Figure 1 shows the graphical representation of the PTN modelling Example 1. $S(X)$ and $E(X)$ stand for the start and the end of a durative event X .

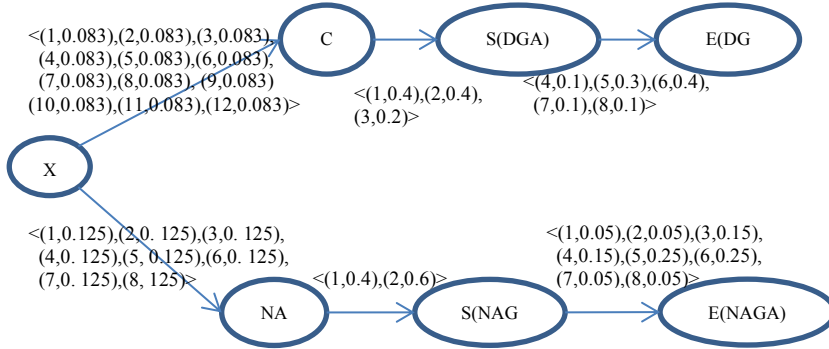


Figure 1. PTN of Example 1.

3.1 Temporal reasoning on PTNs

Our representation model is basically an extension of STP [9] (considering discrete values for the distances) to include probabilities. We can thus perform STP-like

temporal reasoning, adopting Floyd-Warshall's algorithm. However, we had to adapt it to apply to PTNs, by properly instantiating the operators \cap and $@$ in the algorithm Floyd-Warshall's algorithm with two new operators (\cap^p and $@^p$) operating not only on distances, but also on probabilities. Considering distances only, both our intersection and composition operators work as the STP operators. However, they also evaluate the probabilities of the output distances. For technical (computational complexity) reasons, we assume the probabilistic independence of the constraints.

The operator *intersection* \cap^p is used to "merge" two constraints $C1 = \langle (d_1, p^1), \dots, (d_n, p^n) \rangle$ and $C2 = \langle (d'_1, p'^1), \dots, (d'_m, p'^m) \rangle$ concerning the same pair of time points. The set intersection between the two input sets of distances is computed as in STP. The probabilities of each distance belonging to both input constraints are multiplied, and the resulting probabilities are then normalized to sum-up to 1. The formal definition is given below.

Definition 3. Intersection (\cap^p). Given two PQTCs $C1 = \langle (d_1, p^1), \dots, (d_n, p^n) \rangle$ and $C2 = \langle (d'_1, p'^1), \dots, (d'_m, p'^m) \rangle$ their intersection is defined as follows:

let $\{d''_1, \dots, d''_k\} = \{d_1, \dots, d_n\} \cap \{d'_1, \dots, d'_m\}$, then

$$C1 \cap^p C2 = \text{Normal}(\langle d''_1, P^{C1}(d''_1) \cdot P^{C2}(d''_1) \rangle, \dots, \langle d''_k, P^{C1}(d''_k) \cdot P^{C2}(d''_k) \rangle)$$

where $P^{C1}(d)$ and $P^{C2}(d)$ represent the probability of the distance d in the constraint $C1$ and $C2$ respectively, and $\text{Normal}(\langle (d_1, p_1), \dots, (d_n, p_n) \rangle) = \langle (d_1, p_1 / (p_1 + \dots + p_n)), \dots, (d_n, p_n / (p_1 + \dots + p_n)) \rangle$ ■

Example 2. $NAA \langle (2,0.02), (3,0.05), (4,0.09), (5,0.15), (6,0.19), (7,0.25), (8,0.17), (9,0.05), (10,0.03) \rangle \cap NAA \langle (8,0.1), (9,0.2), (10,0.4), (11,0.3) \rangle \rightarrow NAA \langle (8,0.132), (9,0.775), (10,0.093) \rangle$ ■

The composition operator $@^p$ is used to infer the constraint between two time points t and t' , given the constraint $C1$ between t and t_1 and the constraint $C2$ between t_1 and t' . As in STP, output distances are evaluated as the pairwise sums of the input distances. Composition produces all the possible combinations of distances taken from the involved constraints. For each given combination of distances we multiply the corresponding probabilities; the probability of each output distance is the sum of the probabilities of the combinations generating such a distance. More formally:

Definition 4. Composition ($@^p$). Given two PQTCs $C1 = \langle (d_1, p^1), \dots, (d_n, p^n) \rangle$ and $C2 = \langle (d'_1, p'^1), \dots, (d'_m, p'^m) \rangle$, their composition is defined as follows:

let D denote $\{d_1, \dots, d_n\}$ and D' denote $\{d'_1, \dots, d'_m\}$, let $\{d''_1, \dots, d''_r\} = \{d'' : d'' = d_i + d'_j \wedge d_i \in D \wedge d'_j \in D'\}$, and let $p'' = \sum_{d \in D, d' \in D' : d+d'=d''} (P^{C1}(d) \cdot P^{C2}(d'))$, then

$$C1 @^p C2 = \langle (d''_1, p''_1), \dots, (d''_r, p''_r) \rangle,$$

where $P^{C1}(d)$ and $P^{C2}(d')$ represent the probability of the distance d and d' in the constraint $C1$ and $C2$ respectively. ■

Example 3. For example, the composition of the constraint between NAA and the start of NAGA with the one between the start and the end of NAGA gives as result the constraint between NAA and the end of NAGA:

$NAA \langle (1,0.4), (2,0.6) \rangle \cap S(NAGA) @ S(NAGA) \langle (1,0.05), (2,0.05), (3,0.15), (4,0.15), (5,0.25), (6,0.25), (7,0.05), (8,0.05) \rangle \rightarrow NAA \langle (2,0.02), (3,0.05), (4,0.09), (5,0.15), (6,0.19), (7,0.25), (8,0.17), (9,0.05), (10,0.03) \rangle$ ■

Finally, temporal reasoning on a PTN is achieved by Floyd-Warshall's algorithm in Figure 1, in which \cap and $@$ are replaced by our operators \cap^p and $@^p$ respectively. It computes the minimal network, i.e., the strictest temporal constraints (and their probabilities) between each pair of temporal entities (nodes in the PTN).

Example 4. In the *minimal network*, the constraint between S(DGA) and S(NAGA) is: S(DGA) \langle (-13,0.000003), (-12,0.000006), (-11,0.0005), (-10,0.00218), (-9,0.0067), (-8,0.01618), (-7,0.0328), (-6,0.05775), (-5,0.08877), (-4,0.11784), (-3,0.1353), (-2,0.13862), (-1,0.12964), (0,0.10739), (1,0.07651), (2,0.04689), (3,0.025239), (4,0.01167), (5,0.00444), (6,0.00127), (7,0.000234), (8,0.00002) \rangle E(NAGA). ■

4 Querying probabilistic temporal constraints

We provide users with facilities to query the minimal network. We propose the syntax of our query language in *Backus-Naur Form* (augmented with the meta-symbol \cdot to denote non-empty lists), and then we describe our query answering mechanism.

The basic entities on which query operates are events E. They may be instantaneous (E_i ; e.g., NAA) or durative (E_d ; in such a case, they are started and ended by an instantaneous event – e.g., S(DGA), E(DGA)).

$E ::= E_d \mid E_i$

Queries may concern *qualitative relations* (R) between such events. Since we consider both instantaneous and durative events, we consider the relations in Vilain's algebra [17], which include Allen's relations (R_a), but also relations between instantaneous events (R_i), and relations between instantaneous and durative (R_{id}). We add the relation INTERSECT, which is important in the interaction detection task.

$R ::= R_i \mid R_{id} \mid R_a$

$R_i ::= \langle \mid = \mid > \mid \leq \mid \geq \mid \neq \mid \Leftrightarrow \mid \rangle$

$R_{id} ::= \bullet \text{BEFORE} \mid \bullet \text{STARTS} \mid \bullet \text{DURING} \mid \bullet \text{ENDS} \mid \bullet \text{AFTER}$

$R_a ::= \text{BEFORE} \mid \text{MEETS} \mid \text{OVERLAPS} \mid \text{ENDED-BY} \mid \text{CONTAINS} \mid$
 $\text{STARTS} \mid \text{EQUAL} \mid \text{STARTED-BY} \mid \text{DURING} \mid \text{ENDS} \mid$

$\text{OVERLAPPED-BY} \mid \text{MET-BY} \mid \text{AFTER} \mid \text{DISJOINT} \mid \text{INTERSECTS}$

We support both “simple” (Q_s) and hypothetical (Q_h) queries. In turn, “simple” queries can be divided into (i) extraction (Q_e), (ii) qualitative probabilistic (Q_p), and (iii) Boolean probabilistic (Q_b) queries.

$Q ::= Q_s \mid Q_h$

$Q_s ::= Q_e \mid Q_p \mid Q_b$

Extraction queries. Trivially, given a set of pairs of events, such queries give as output the probabilistic temporal constraints between each pair, taken from the minimal network.

$Q_e ::= \{ \langle E, E \rangle \} ?$

Example 5. For example, the query (Q2) asks for the temporal constraints (and their probabilities) between the start of DGA and the start of NAGA.

(Q2): $\{<S(DGA), S(NAGA)\}$? ■

Qualitative probabilistic queries. They ask the probability of a qualitative temporal relation between two events.

$$Q_r ::= \text{prob}(A_r)?$$

$$A_r ::= E_i R_i E_j \mid E_i R_{ij} E_j \mid E_i R_i E_j$$

Example 6. Physicians can ask (given the constraints in Ex.1) what is the probability that the effects of CCA and NAA intersect in time (i.e., what is the probability of the interaction between CCA and NAA) through the query (Q3).

(Q3) $\text{Prob}(DGA(\text{INTERSECT})NAGA)$. ■

Boolean probabilistic queries. These queries ask whether the probability of a qualitative relation A_r (as above) is $<$, $>$, $=$, \leq , \geq or \neq with respect to a given probability P .

$$Q_b ::= \text{Prob}(A_r) \text{ Op } P ?$$

Example 7. The query (Q4) asks whether the probability that DGA starts before NAGA are greater than 0.5.

(Q4) $(\text{Prob}(S(DGA) > S(NAGA)) > 0.5)$ ■

Hypothetical queries. Such queries are “simple” (i.e., extraction, qualitative probabilistic or Boolean probabilistic) queries to be answered in the context in which a set of PQTCs (denoted by C in the BNF below) is assumed.

$$Q_h ::= Q_r \text{ if } \{C\}$$

Example 8. The query (Q1) in the introduction can be expressed as:

(Q1') $\text{Prob}(DGA(\text{INTERSECT})NAGA) \text{ if } \{X_{<(1,0.5),(2,0.5)>CAA}, X_{<(1,0.5),(2,0.5)>NAA}\}$ ■

4.1 Query evaluation

The minimal network of the PTN (henceforth MN) must be available to answer queries. Thus, if it is not available, it must be computed, as discussed in Section 3.

(1) **Extraction queries.** Given the MN, such queries can be answered by returning to the user the constraints in the MN concerning the events specified in the query.

Example 9. The output of query Q2 is the constraint shown in Example 4. ■

(2) **Qualitative probabilistic queries.** To evaluate a probabilistic query, we first have to define the probabilities of the relationships between instantaneous events.

Given two any instantaneous events e_i and e_j , and given the temporal constraint $e_i <(d,p), \dots, (d,p)> e_j$, we indicate with $\varphi(d)$ the probability of the distance d (i.e., $\varphi(d)=p$). The probabilities are evaluated as below. For example, since $e_i <(d,p), \dots, (d,p)> e_j$ states that the possible distances of e_j with respect to e_i are d_1, \dots, d_n , e_i precedes e_j for all the distances $d_i \in \{d_1, \dots, d_n\}$ such that $d_i > 0$. Therefore the probability $\text{Prob}(e_i > e_j)$ is the sum of the probabilities of such distances.

Definition 5. Given a constraint $e_i <(d,p), \dots, (d,p)> e_j$

$\text{Prob}(e_i > e_j) = \sum_{d_i > d_j} \varphi(d_i)$ if $\exists d_i \in \{d_1, \dots, d_n\}$ such that $d_i > d_j$ (0 otherwise)

$\text{Prob}(e_i = e_j) = \varphi(0)$ if $0 \in \{d_1, \dots, d_n\}$ (0 otherwise)

$\text{Prob}(e_i < e_j) = \sum_{d_i < d_j} \varphi(d_i)$ if $\exists d_i \in \{d_1, \dots, d_n\}$ such that $d_i < d_j$ (0 otherwise) ■

Example 10. Given the MN in Example 4, $\text{Prob}(S(\text{DGA}) > S(\text{NAGA})) = 0.166273$. ■

The probabilities of “ambiguous” relationships between instantaneous events can be simply evaluated on the basis of the definition above, as the sum of the probabilities of the alternative basic relationships that constitute them.

Definition 6.

$\text{Prob}(t_i \geq t_j) = \text{Prob}(t_i > t_j) + \text{Prob}(t_i = t_j)$

$\text{Prob}(t_i \neq t_j) = \text{Prob}(t_i > t_j) + \text{Prob}(t_i < t_j)$

$\text{Prob}(t_i \leq t_j) = \text{Prob}(t_i < t_j) + \text{Prob}(t_i = t_j)$ ■

The probabilities of atomic temporal relations between two durative events e_i and e_j can consequently be evaluated as shown in Definition 7 (the probabilities of the qualitative relations between an instantaneous and a durative event can be defined in a similar way, and are omitted for the sake of brevity).

Definition 7.

$\text{Prob}(e_i(\text{BEFORE})e_j) = \text{Prob}((\text{end}(e_i)+1) < \text{start}(e_j))$

$\text{Prob}(e_i(\text{MEETS})e_j) = \text{Prob}((\text{end}(e_i)+1) = \text{start}(e_j))$

$\text{Prob}(e_i(\text{OVERLAPS})e_j) = \text{Prob}(\text{start}(e_i) < \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) \geq \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) < \text{end}(e_j))$

$\text{Prob}(e_i(\text{ENDED-BY})e_j) = \text{Prob}(\text{start}(e_i) < \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) = \text{end}(e_j))$

$\text{Prob}(e_i(\text{CONTAINS})e_j) = \text{Prob}(\text{start}(e_i) < \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) > \text{end}(e_j))$

$\text{Prob}(e_i(\text{STARTS})e_j) = \text{Prob}(\text{start}(e_i) = \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) < \text{end}(e_j))$

$\text{Prob}(e_i(\text{EQUAL})e_j) = \text{Prob}(\text{start}(e_i) = \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) = \text{end}(e_j))$

$\text{Prob}(e_i(\text{STARTED-BY})e_j) = \text{Prob}(\text{start}(e_i) = \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) > \text{end}(e_j))$

$\text{Prob}(e_i(\text{DURING})e_j) = \text{Prob}(\text{start}(e_i) > \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) < \text{end}(e_j))$

$\text{Prob}(e_i(\text{ENDS})e_j) = \text{Prob}(\text{start}(e_i) > \text{start}(e_j)) \cdot \text{Prob}(\text{end}(e_i) = \text{end}(e_j))$

$\text{Prob}(e_i(\text{OVERLAPPED-BY})e_j) = \text{Prob}(\text{start}(e_i) > \text{start}(e_j)) \cdot \text{Prob}(\text{start}(e_i) \leq \text{end}(e_j)) \cdot \text{Prob}(\text{end}(e_i) > \text{end}(e_j))$

$\text{Prob}(e_i(\text{MET-BY})e_j) = \text{Prob}((\text{end}(e_i)+1) = \text{start}(e_j))$

$\text{Prob}(e_i(\text{AFTER})e_j) = \text{Prob}((\text{end}(e_i)+1) < \text{start}(e_j))$

$\text{Prob}(e_i(\text{INTERSECTS})e_j) = \text{Prob}(\text{start}(e_i) \leq \text{start}(e_j)) \cdot \text{Prob}(\text{start}(e_i) \leq \text{end}(e_j)) + \text{Prob}(\text{start}(e_i) \leq \text{start}(e_j)) \cdot \text{Prob}(\text{start}(e_j) \leq \text{end}(e_i)) - \text{Prob}(\text{start}(e_i) \leq \text{start}(e_j)) \cdot \text{Prob}(\text{start}(e_i) \leq \text{end}(e_j)) \cdot \text{Prob}(\text{start}(e_j) \leq \text{end}(e_i))$. ■

Example 11. Given the MN for the constraints in Example 1, $\text{Prob}(\text{DGA}(\text{DURING})\text{NAGA})=0.085$, and $\text{Prob}(\text{DGA}(\text{INTERSECT})\text{NAGA})=0.995$ ■

(3) **Boolean probabilistic queries.** Given the MN, such queries can be answered by evaluating the probability of the qualitative relation (as above), and comparing it with the probability in the query.

(4) **Hypothetical queries.** To answer hypothetical queries:

- (1) First, the “hypothesized” temporal constraints in $\{C\}$ are added to the MN (through the intersection with previous temporal constraints; see Section 3.1)
- (2) Temporal reasoning is performed (through Floyd-Warshall’s algorithm), producing a new MN
- (3) The probability of the conditions (left part of the query) is evaluated in the new MN, as discussed above.

Example 11. The evaluation of the query (Q4) above requires the addition of the constraints $\{X_{i,j} < (1,0.5), (2,0.5) > CAA, X_{i,j} < (1,0.5), (2,0.5) > NAA\}$ into the MN, and a new propagation. The result of the query is: **0.9943**.

5 Probabilistic temporal detection of interactions

Though our temporal approach is domain-independent, we designed it with specific attention to the GLARE application. When executing multiple guidelines on a comorbid patient, physicians can adopt GLARE’s facilities to study possible interactions between treatments. During the acquisition phase the temporal constraints and their probabilities are acquired jointly by expert physicians and knowledge engineers. Several aspects influence the absorption of a drug and its effects. Pharmacokinetics and pharmacodynamics study such mechanisms. It integrates a pharmacokinetic and a pharmacodynamic model component into a set of mathematical expressions that allows the description of the effect intensity in time wrt the administration of a drug dose. Deriving from such mathematical expressions the probabilities of the effects of a drug along time could be difficult. As a first approximation, we considered the models of the plasma concentrations of the drugs, their half-life (i.e., the time to halve the drug amount in the blood) and the type of the effect, and we have approximated the probabilities with the help of medical experts.

At each time during the execution, physicians can trigger GLARE’s interaction analysis mechanism to check whether interactions may arise among the next actions to be executed in the guidelines. Probabilistic temporal reasoning is used to check not only whether interactions are temporally possible, but also their probabilities. The output of temporal reasoning is a complex network of PQTCs. For example, in the case of our running example, we have a set of constraints like the one in Example 4 above, one for each pair of instantaneous events (or starting/ending points of durative events). Obviously, the MN is hard to understand. Thus, we consider our query language an essential support for physicians. To facilitate them, we also provide a graphical interface, which makes the formulation of queries more user-friendly. Indeed, the physicians working in the GLARE project asked us for a temporal support to cope with two main situations:

- (1) *They are already executing one or more therapies on a patient. Focusing on the next actions, they analyze whether interactions are temporally possible.*
- (2) *They are going to choose among alternative therapies in a guideline, and they want to analyze the alternatives to check whether they may interact with the other therapies currently in execution for the patient.*

Situation (1). In such a context, queries in general, and “INTERSECTS” queries (hypothetical or not) in particular, are very helpful. Notably, probabilities are very

important, since physicians tend to accept interactions having low probabilities (indeed, all drugs, even considered in isolation, have a list of –not highly probable– undesirable side effects). In such a context, also hypothetical temporal queries are very useful: physicians exploit such a facility to check whether they can decrease the probabilities of interactions by executing actions at “proper” times. In our running example, the query Q1 can be expressed by physicians (through a graphical interface) as Q1’ in Example 8, and the output would be the probability **0.9943**. Given the high probability, physicians may still try to see whether, choosing specific execution times for some actions, such a probability can be decreased. For example, physicians might ask a query like Q5 (to check the probability of interaction in case NAA is executed in the first 30 minutes, and CAA between two and three hours from the current time):

(Q5) **Prob(DGA(INTERSECTS)NAGA)**

if {X,<(1,0.5),(2,0.5)>NAA, X,<(9,0.25),(10,0.25),(11,0.25),(12,0.25)>CAA}

The output probability is **0.02455**, suggesting to the physicians that the probability of interaction sharply decreases if they delay the execution of CAA. Notably, using “crisp” temporal constraints, physicians could only know that an interaction may occur, both in case CAA is executed within the first 30 minutes, and in case it is executed after two or three hours.

Situation (2). From the point of view of our support, situation (2) is similar to situation (1) above. Simply, physicians have to iterate the checking process on each one of the alternatives that they think can be appropriate for the patient at hand.

6 Conclusions and Future Work

Dealing and reasoning with temporal information in CIGs is an important issue [2]. In our previous works we coped with temporal reasoning problems and in particular with temporal indeterminacy in the areas of CIGs [6, 18] and relational databases [19, 20]. In this paper, we have proposed the first approach for reasoning and query answering about probabilistic quantitative temporal constraints, and its application within the GLARE project, for the analysis of the interactions between guidelines. Preliminary tests conducted with the physicians cooperating in the GLARE project show that they appreciate a probabilistic approach (with respect to “traditional” “crisp” approaches, that can only say whether an interaction is certain, possible or impossible). The development of “physician-friendly” graphical facilities to acquire, treat and query probabilistic temporal constraints, and a more extensive evaluation with other physicians are two of the main goals of our future work.

References

1. Ten Teije, A., Miksch, S., Lucas, P. eds: Computer-based medical guidelines and protocols: a primer and current trends. IOS Press, Amsterdam (2008).

2. Peleg, M.: Computer-interpretable clinical guidelines: A methodological review. *Journal of Biomedical Informatics*. 46, 744–763 (2013).
3. Terenziani, P., Molino, G., Torchio, M.: A modular approach for representing and executing clinical guidelines. *Artificial Intelligence in Medicine*. 23, 249–276 (2001).
4. Bottrighi, A., Terenziani, P.: META-GLARE: A meta-system for defining your own computer interpretable guideline system - Architecture and acquisition. *Artificial Intelligence in Medicine*. 72, 22–41 (2016).
5. Piovesan, L., Anselma, L., Terenziani, P.: Temporal detection of guideline interactions. In: *Proc. HEALTHINF 2015, Part of BIOSTEC 2015*. pp. 40–50 (2015).
6. Anselma, L., Piovesan, L., Terenziani, P.: Temporal detection and analysis of guideline interactions. *Artificial intelligence in medicine*. 76, 40–62 (2017).
7. Terenziani, P., Andolina, A.: Probabilistic quantitative temporal reasoning. In: *Proc. Symposium on Applied Computing*. pp. 965–970. ACM (2017).
8. Schwalb, E., Vila, L.: Temporal constraints: A survey. *Constraints*. 3, 129–149 (1998).
9. Dechter, R., Meiri, I., Pearl, J.: Temporal Constraint Networks. *Artif. Intell.* 49, 61–95 (1991).
10. Badaloni, S., Giacomini, M.: The algebra IAFuz: a framework for qualitative fuzzy temporal reasoning. *Artificial Intelligence*. 170, 872–908 (2006).
11. Ryabov, V., Trudel, A.: Probabilistic temporal interval networks. In: *Proc. TIME 2004*. pp. 64–67. IEEE (2004).
12. Mouhoub, M., Liu, J.: Managing uncertain temporal relations using a probabilistic interval algebra. In: *Proc. IEEE Int Conf on Systems, Man and Cybernetics*. pp. 3399–3404. (2008).
13. Khatib, L., Morris, P., Morris, R., Rossi, F.: Temporal constraint reasoning with preferences. In: *Proc. IJCAI 2001*. pp. 322–327. Morgan Kaufmann (2001).
14. Terenziani, P., Andolina, A., Piovesan, L.: Managing Temporal Constraints with Preferences: Representation, Reasoning, and Querying. *IEEE Trans. Knowl. Data Eng.* 29, 2067–2071 (2017).
15. Zamborlini, V., Da Silveira, M., Pruski, C., ten Teije, A., van Harmelen, F.: Towards a conceptual model for enhancing reasoning about clinical guidelines. In: *Workshop on Knowledge Representation for Health-Care*. pp. 29–44. Springer (2014).
16. Piovesan, L., Terenziani, P.: A Constraint-Based Approach for the Conciliation of Clinical Guidelines. In: *IBERAMIA 2016*. pp. 77–88. Springer (2016).
17. Vilain, M.: A System for Reasoning About Time. In: Waltz, D.L. (ed.) *Proc. AAAI 82*. Pittsburgh, PA, August 18-20, 1982. pp. 197–201. AAAI Press (1982).
18. Anselma, L., Bottrighi, A., Montani, S., Terenziani, P.: Managing proposals and evaluations of updates to medical knowledge: theory and applications. *J Biomed Inform.* 46, 363–376 (2013).
19. Anselma, L., Piovesan, L., Terenziani, P.: A 1NF temporal relational model and algebra coping with valid-time temporal indeterminacy. *Journal of Intelligent Information Systems*. 47, 345–374 (2016).
20. Anselma, L., Stantic, B., Terenziani, P., Sattar, A.: Querying now-relative data. *Journal of Intelligent Information Systems*. 41, 285–311 (2013).