Temporal reasoning and query answering with preferences and probabilities for medical decision support

Please, cite this paper as:
Antonella Andolina, Marco Guazzone, Luca Piovesan, Paolo Terenziani
“Temporal reasoning and query answering with preferences and probabilities for medical decision support,”
Expert Systems with Applications, Volume 195, 2022, 116565.
DOI: 10.1016/j.eswa.2022.116565
Publisher: https://doi.org/10.1016/j.eswa.2022.116565

Temporal reasoning and query answering with preferences and probabilities for medical decision support

Antonella Andolina[1], Marco Guazzone[2,3], Luca Piovesan[2,3*], Paolo Terenziani[2,3]

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

[2]DISIT, Università del Piemonte Orientale, Viale Teresa Michel 11, 15121 Alessandria, Italy

[3]AI@UPO, Università del Piemonte Orientale, Vercelli, Italy

antoando@libero.it, {marco.guazzone, luca.piovesan, paolo.terenziani}@uniupo.it

[*] Corresponding author, phone number: +39 0131 360 180

Abstract

Knowledge-based decision support systems have a long tradition within the medical area. In particular, in the last decades, many Computer-Interpretable Guidelines (CIG) systems have been built to provide evidence-based and knowledge-based support to physicians. Since CIGs are, by definition, devoted to the management of specific diseases, the treatment of comorbid patients constitutes a challenging task in the area, involving (i) the detection of the possible interactions between (the effects of) the actions recommended by multiple CIGs (one for each disease of the patient), (ii) the management of such interactions and, finally, (iii) the conciliation of (the recommendations of) different CIGs. This paper focuses on issue (i) above, and specifically, on an innovative approach to support interaction detection along the temporal dimension. Practically, interactions can only occur between effects that *intersect* in time. Therefore, interaction detection involves the *representation* of temporal information (temporal constraints), and temporal *reasoning* (to *propagate* such constraints). Additionally, query answering facilities are important to support physicians in the investigation of the results of temporal reasoning. Current CIG approaches that face such issues take into account only "crisp" temporal constraints, i.e., they consider all temporal constraints as equally probable\preferred. However, *preferences* about the temporal constraints between CIGs actions may be available, as well as knowledge about the *probabilistic* distribution of the effects of CIGs actions in time. Considering such additional pieces of information can provide crucial advantages, in term of the flexibility and informativeness of the support provided by the CIG system to physicians. In this paper, we propose the first homogeneous approach to represent and reason with (propagate) ***temporal constraints with both preferences and probabilities***. We ground our approach on the widely-used Simple Temporal Problem (STP) framework, which supports temporal reasoning on temporal constraints about possible distances between events. We extend (i) the representation formalism to associate preferences and\or probabilities to the possible distances, and (ii) the operations to propagate the constraints to combine also preferences and probabilities. We also (iii) provide an experimental evaluation of our approach, and (iv) propose a wide range of query-answering supports, to facilitate physicians in the analysis of the results of temporal reasoning in general, and in the temporal detection of possible interactions in particular. Finally, (v) we also show how such a temporal framework is integrated in GLARE-SSCPM, a CIG system to treat comorbid patients, and show the advantages of our approach considering a running example.

KEYWORDS: Knowledge-based decision support; Clinical guideline interaction detection; Temporal reasoning; Probabilities; Preferences; Query answering

1 INTRODUCTION

Knowledge-based decision support systems have a long tradition within the medical area. In particular, in the last decades, many AI approaches have been developed to provide *evidence-based* and *knowledge-based* decision support to physicians, based on *Clinical Practice Guidelines* (*CPGs*). CPGs are – in the definition of the American Institute of Medicine – "systematically developed statements to assist practitioner and patient decisions about appropriate health care in specific clinical circumstances" (Institute of Medicine, Committee on Quality Health Care in America, 2001). They are large bodies of knowledge supporting physicians in an evidence-based, knowledge-based, standardized, and optimized treatment of patients. A lot of CPGs, developed by national and\or international organizations are publicly available (e.g., the Guideline International Network (Guidelines International Network, n.d.) provides more than 6500 CPGs). In the last twenty years, many computer-based approaches have been developed to manage *Computer-Interpretable Guidelines* (henceforth *CIGs*; consider, e.g., the surveys (Peleg, 2013; Ten Teije et al., 2008)). The adoption of computer-based ap-

proaches to manage CIGs provides crucial advantages with respect to the direct use of traditional textual CPGs: in particular, the automatic connection to the patient's electronical record allows CIG system to provide *patient-specific* recommendations\decision support to physicians. As a consequence, many CIG systems have been built. While commercial CIG systems are mostly domain-specific (i.e., they are devoted to the treatment of a specific guideline, for a specific disease), a large stream of research in *Artificial Intelligence in Medicine* (*AIM*) is devoted to the development of domain-independent CIG systems. Such systems usually resemble "old-style" medical expert systems, providing physicians with an *acquisition* tool to acquire medical knowledge (i.e., a specific CPG) in an internal format, and an *execution* tool, which can be conceived as a *decision support tool*, supporting physicians in the application of an acquired CIG to a specific patient.

Given also the increasing ageing of the population, the treatment of ***comorbid patients*** (i.e., patients affected by more than one disease) is a hot problem, which is attracting a lot of attention in AIM. By definition (see, e.g., (Institute of Medicine, Committee on Quality Health Care in America, 2001)), each CPG (and CIG) focuses on a specific disease only. Though ad-hoc CPGs can be devised to consider the most common cases of co-occurrence of diseases, in the general case it is not possible to imagine the definition of a new CPG for each possible combination of diseases, so that new methods have to be devised. The development of such methods has been identified as one of the "grand challenges" for clinical decision support (Sittig et al., 2008).

In most cases, coping with comorbid patients involves the need of combining multiple CPGs (CIGs). And, unfortunately, the effects of the recommendations in different CIGs may interact, and some of such interactions may be (very) dangerous for the health of the patient. To face such a challenging problem, in the last decade many AIM approaches have started to devise computer-based approaches which extend CIG frameworks to provide physicians with decision support capabilities also in case of comorbid patients. A general overview of the different AIM CIG-based approaches coping with comorbidities in the literature would lead us far away from the main goals of this paper (consider, among the others, (Jafarpour et al., 2019; Jafarpour & Abidi, 2013; Kogan et al., 2018; Merhej et al., 2016; Michalowski et al., 2021; Piovesan et al., 2018; Piovesan & Terenziani, 2016; Riaño & Collado, 2013, p. 13; Sánchez-Garzón et al., 2013; Wilk et al., 2017; Zamborlini et al., 2017; Zhang & Zhang, 2014) and the recent surveys in (Bilici et al., 2019; Fraccaro et al., 2015; Riaño & Ortega, 2017)). As an example of such approaches, in Section 3 of this paper we briefly introduce GLARE-SSCPM (GLARE Support System for Comorbid Patient Management), which constitutes the framework in which the approach proposed in this paper is integrated. For the sake of this paper, it is only important to highlight that, in the CIG approaches in the literature, the problem of coping with comorbidities is usually split into three main and independent subproblems:

(i) knowledge-based detection of possible interactions between CIGs,
(ii) management (i.e., resolution) of the interactions, and
(iii) conciliation of the interacting CIGs (i.e., merging the recommendations of multiple CIGs in such a way that interaction managements are considered).

A large literature has been devoted to each one of the above topics. In this paper, we focus on subproblem (i), and, more specifically, on the *temporal issues* it involves. In general, temporal knowledge (often expressed in the form of ***temporal constraints***) and ***temporal reasoning*** are fundamental in many medical tasks, including decision support (Adlassnig et al., 2006; Augusto, 2005).

In particular, temporal constraints are fundamental in the CPG and CIG context (consider, e.g., temporal constraints about when the different actions should be executed, about the required delay between actions, or the periodicity of a treatment such as a chemotherapy), in which multiple types of temporal information are involved (consider, e.g., the survey in (Terenziani et al., 2008)).

As a consequence, many CIG formalisms in the literature support the specification of temporal constraints (see, e.g., the survey (Terenziani et al., 2008), and Section 2.3 of this paper). In particular, *temporal constraints are very important when considering multiple CIGs to deal with comorbid patients*. Indeed, knowledge about the effects of CIGs' actions and about the possible interplay between such effects may be exploited in order to detect the possibility of interactions between CIGs (see, e.g., (Zamborlini et al., 2017) and GLARE-SSCPM (Piovesan et al., 2018)). However, only a temporal analysis concerning ***when*** actions have to be executed and ***when*** their effects can occur can discriminate between those interactions that ***may effectively occur in time***, and the ones that cannot. Indeed, interactions (between the effects of CIG actions) may occur only in case effects may ***intersect in time***. Therefore, detecting *temporally possible interactions* is a challenging task, requiring temporal knowledge (in the form of ***temporal constraints*** between CIG actions, and between actions and their effects) and *temporal reasoning* (e.g., in the form of ***temporal constraint propagation***). Some of the current AIM approaches in the literature face such a problem, taking advantage of general frameworks, that natively support representation of and reasoning about constraints, including temporal ones (con-

sider, e.g., (Jafarpour et al., 2019; Michalowski et al., 2021; Piovesan et al., 2020; Wilk et al., 2017)), other approaches like GLARE and GLARE-SSCPM (Anselma et al., 2017; Piovesan et al., 2018) propose a *specialized framework to model and propagate temporal constraints*, and integrate it in the overall CIG system (see the discussion in Section 2.3 below).

Though interesting and useful, the above CIG *temporal* approaches share a common limitation: they deal only with "***crisp***" temporal constraints, i.e., they consider all temporal constraints as equally probable\preferred. Unfortunately, "crisp" approaches are often too rigid, since "crisp" constraints may be either satisfiable or not. Indeed, this may be a severe limitation in all cases in which preferences and\or probabilities are (directly or indirectly) available. Specifically, in the CIG context, several guidelines provide not only temporal constraints about the delay between actions, but also indicate that different delays have different *preferences*. A typical example is the administration of antibiotics, in which the delays between administrations depend on pharmacokinetics factors: the delays with maximum preference are the ones that guarantee that the level of the drug in the patient remains as much as possible constant, and preferences decreases while moving away from such preferred delays. Preferences for CIG actions can be also derived from patient's preferences (consider, e.g., the time of administration of drugs) or from organizational reasons (e.g., of an hospital). Additionally, *probabilistic* knowledge about the time and duration of the effects of CIG actions may be available\derived. Consider again, for instance, drug administrations. Pharmacokinetics provides mathematical models describing the time course of drug absorption, distribution, metabolism, and excretion in a specific patient, while pharmacodynamics provides models describing the time course and intensity of therapeutic effects (Spruill et al., 2014). Such models are the result of studies carried out both in vivo, in vitro and, in the last years, in silico (i.e., through computer-based simulators – see, e.g., (Burghaus et al., 2014; Johnstone et al., 2017) and the survey (Ekins et al., 2007)). Some approaches in literature have started to integrate pharmacokinetics (drug concentration vs time) and pharmacodynamics (effect vs drug concentration) models obtaining integrated models quantitatively and probabilistically describing the course of an effect of a specific drug in time[1] (see, e.g., (Derendorf & Meibohm, 1999; Mehrotra et al., 2007)).

Notably, in the CIG context, *both preferences and probabilities* may be available. The former regard the (temporal constraints between) CIG actions, whose execution is under the control of physicians. The latter regards the (time of occurrence of) the effects of CIG actions. The time of the rise of effects is not under the control of physicians, so that, in this case, we have probabilities (of occurrence at a given time), and not preferences (which involves an agent's choice). As a concrete example, let us consider Ex.1.

Ex.1. Let us consider a comorbid patient suffering from Gastroesophageal reflux disease (GERD) and from Urinary Tract Infection (UTI). Among the other treatments, the clinical guideline for GERD recommends Calcium Carbonate Administration (CCA), to be dispensed when needed. Assuming units of 15 minutes as temporal granularity, and considering preferences in the [0,1] real interval, with 0 and 1 denoting the minimum and maximum preferences, respectively, CCA can be administered 0 or 1 units (i.e., between 0 and 15 minutes) after the decision about the reflux symptoms (RS) with preference 1, 2 or 3 units after RS with preference 0.75, 4 or 5 units after RS with preference 0.5, and finally 6 units after RS with preference 0.25. Among the other effects, CCA leads to the Decreasing Gastric Absorption (DGAb), which can begin after 1, 2, or 3 units, with probability 0.4, 0.4 and 0.2 respectively. In addition, DGAb can end 5, 6, 7, 8, 9 or 10 units after CCA, with probability 0.1, 0.2, 0.3, 0.2, 0.1 and 0.1 respectively. We suppose that the patient is currently treated with (orally administered) Nalidixic Acid, as recommended by the clinical guideline for UTI. Each Nalidixic Acid Administration (NAA) should be dispensed exactly six hours (i.e., 24 time units) after the previous one with preference 1. The preference value decreases by moving (before or after) such a maximally preferred time: 20 units after has preference 0.25, 21 has preference 0.5, 22 has preference 0.75, 23, 24 and 25 have preference 1, 26 has preference 0.75, 27 has preference 0.5 and finally performing NAA 28 units after the previous one has preference 0.25. The Increase of Nalidixic Acid Gastric Absorption (INAGA) begins after 1 unit from NAA with probability 0.4, or 2 units with probability 0.6. INAGA can end 2, 3, 4, 5, 6, 7 or 8 units after NAA, with probability 0.15, 0.25, 0.25, 0.15, 0.10, 0.05 and 0.05 respectively. ■

Notably, preferences regard the time of execution of drug administrations (i.e., actions which are under the control of physicians), while probabilities regard the general knowledge about the delay and duration of effects (which are not under the control of physicians).

Note.1. *Preferences are evaluated in each guideline independently of the other guidelines (e.g., they do not consider the possibility that multiple guidelines can be executed on a patient).*

---

[1] Notably, even if online repositories (e.g., drugbank (Wishart et al., 2018)) provide, at least in plain text, both pharmacokinetics and pharmacodynamics information, they do not provide information about integrated models. To the best of our knowledge, a repository containing such a knowledge still does not exist. For such a reason, the probabilities of the drug effects used in this paper have been defined by an expert by using pharmacokinetics and pharmacodynamics information taken from the specialized literature.

In Ex.1, DGAb and INAGA are potentially interacting (and contrasting) effects, since DGAb may decrease the body's capability of absorbing Nalidixic Acid (NA). In "crisp" approaches to temporal constraints (in which no probability and no preference can be managed), ***temporal reasoning*** can be used to propagate the temporal constraints (e.g., to infer the implied temporal constraints between each endpoint of actions and of their effects), and, after the propagation, ***query answering*** facilities can be provided to check, given the time when CCA and NAA are executed, whether the interaction between DGAb and INAGA must\may temporally occur, or not. However, in the case preferences and probabilities are available, such a response would be quite incomplete. Indeed, in case the interaction may temporally occur, physicians might want to know also what are the temporal probabilities of the interaction, and the temporal preferences in the given temporal scenario. However, to provide them with such additional knowledge, we have to move from "crisp" to "non-crisp" temporal constraints, in which ***both temporal preferences and temporal probabilities*** are considered.

Notably, the limitations of approaches facing only "crisp" constraints are not at all specific of the medical application domain, and have been noticed since a long time by the AI literature. As a consequence, several *domain-independent* approaches have been already devised in order to enrich temporal constraints with preferences or probabilities, and to perform temporal constraint propagation in such an extended context (see the discussion in Section 2.2). However, such approaches take into account *either preferences or probabilities*, while *none of them consider both*, as required in the CIG context (see again Ex.1). Focusing specifically on CIG approaches, "non-crisp" temporal constraints have been considered only in (Andolina et al., 2018). In such an approach, STP temporal constraints have been extended to associate *probabilities* to the temporal distances, and a temporal reasoning algorithm (based on constraint propagation) has been devised to propagate such constraints.

In this paper, we overcome the above general limitation of the AI literature. We go on with the mainstream of AI research overviewed in Section 2, and already followed in the GLARE and GLARE-SSCPM projects (Anselma et al., 2017; Piovesan et al., 2018), by proposing a specialized temporal framework to manage temporal constraints. Specifically, in this paper we propose the *first* temporal framework in the literature able to represent, reason and query about "non-crisp" temporal constraints, and considering *both preferences and probabilities*. This paper is a (very) extended version of our AIME'19 short paper (Terenziani & Andolina, 2019) (five pages, LNCS format). In (Terenziani & Andolina, 2019), we have just proposed the idea of integrating both preferences and probabilities in STP temporal constraints. We have proposed the extended representation formalism, and we have shown an example of application. No technical elaboration of the idea has been proposed in (Terenziani & Andolina, 2019): indeed, the contributions in Sections 5, 6, and 7 of this paper are entirely new.

Specifically, the main contributions of our approach are:

(1) The definition of a formalism to represent quantitative temporal constraints with preferences and probabilities (Section 4).
(2) The definition of the operations (intersection and composition) to propagate such temporal constraints, including their preferences and probabilities (such operations are the core of our temporal reasoning algorithm, and are presented in Section 5).
(3) An experimental evaluation of our extended temporal reasoning approach (Section 5.3).
(4) A rich query language (and its related query-answering support) to query the results of temporal reasoning, with specific support to check temporal intersection (i.e., the temporal possibility of interactions), and to consider hypothetical queries (to support *what-if* reasoning) (Section 6).
(5) The integration of the above contributions into GLARE-SSCPM, to analyze interactions between CIGs (Sections 5.4 and 6.6).

Notably, though in this paper we consider the integration of our temporal framework into GLARE-SSCPM, the temporal approach we propose in Sections 4—5 (and, partly, in Section 6) is task and domain independent, and can operate as a *temporal knowledge server* for different systems, and for different applications.

The paper starts with two background sections. In Section 2, we provide the necessary background about temporal constraints and temporal reasoning. Then, in Section 3, we briefly resume the main features of the GLARE-SSCPM system, which provides physicians with CIG-based decision support for the treatment of comorbid patients.

2 PRELIMINARIES AND RELATED WORKS

In this background section we briefly illustrate some relevant AI approaches to temporal reasoning about "crisp" (Section 2.1) and "non-crisp" (Section 2.2) temporal constraints, and then discuss the treatment of temporal constraints in current CIG systems (Section 2.3).

2.1 "Crisp" temporal constraints and reasoning

In general, and informally, temporal constraints can be interpreted as limitations about when events occur\should occur. In AI, a whole stream of research aims at proposing special-purpose frameworks (usually called *temporal reasoners*) to represent and reason with temporal constraints (see, e.g., the survey in (Vila, 1994)). Such frameworks can be used as specialised knowledge servers to which temporal problems can be demanded, to solve complex tasks (e.g., planning, and scheduling) in an efficient and compositional way. The AI special-purpose approaches on temporal constraints have been traditionally divided into two main classes, depending on whether they deal with *quantitative* or *qualitative* temporal constraints (see, e.g., the surveys in (Barták et al., 2014; Schwalb & Vila, 1998; Terenziani, 2006; Vila, 1994)). ***Quantitative*** temporal constraints involve *metric time* and include ***dates*** (e.g., "Mary was enrolled on 10/1/2020"), ***delays*** (e.g., "Sue was enrolled 15 days after Mary"), and ***durations*** (e.g., "Mary worked for the company XXX for 120 days"). ***Qualitative*** temporal constraints concern the relative position of events (e.g., "John arrived at work after Mary"). Notably, in many cases, temporal constraints are ***not exact*** (e.g., "Sue was enrolled between 20 and 40 days after Mary"). Two famous approaches to *quantitative* constraints are, e.g., (Dechter et al., 1991; Koubarakis, 1997). In particular, in the *Simple Temporal Problem* (*STP*) framework (Dechter et al., 1991), constraints of the form $t_i[d_{min},d_{max}]t_j$ represent the minimum (d_{min}) and maximum (d_{max}) temporal distances between pairs of time points t_i and t_j. Notably, exact and non-exact dates, durations and delays can be easily encoded in STP.

Qualitative approaches consider qualitative temporal constraints, i.e., constraints about the relative temporal ordering of events (e.g., event e_1 occurred before event e_2). For instance, Allen's famous Interval Algebra (Allen, 1983) copes with time intervals, while the Point Algebra (M. B. Vilain & Kautz, 1986) deals with time points, and Vilain's Point-Interval Algebra copes with both points and intervals (M. Vilain, 1982).

Moreover, also hybrid approaches have been developed, considering both quantitative and qualitative temporal constraints (e.g., (Brusoni et al., 1997; Kautz & Ladkin, 1991; Meiri, 1996)). For instance, the Later temporal reasoner is based on the STP framework, and supports, besides quantitative temporal constraints, also those qualitative temporal constraints that can be mapped onto such a framework (Brusoni et al., 1997).

The core of such approaches are ***temporal reasoning*** algorithms to propagate the constraints. Depending on the framework, temporal reasoning is developed in order to achieve different goals: (i) consistency checking, (ii) to find a *solution* (i.e., an instantiation of variables that satisfies all the constraints; also called *scenario*), or (iii) to find out the *minimal network* (Dechter et al., 1991) (i.e., the tightest implied constraints). Notably, in many tasks, such as in *decision support* (*which includes the CIG context we consider in this paper*), providing users with a specific solution does not suffice (physicians want to choose themselves among the different possibilities), and presenting them all the solutions is practically infeasible. In such tasks, a good option is to provide users with (i) the ***minimal network*** (which represents in a compact and implicit way all the possible solutions), and (ii) some ***query answering*** facility, to let them explore the space of solutions, facilitating their choice (consider, e.g., (Brusoni et al., 1997)).

Ex.2. As an example, consider the set *KB* of STP constraints, where t_1, t_2 and t_3 denote time points: $KB=\{t_1[2,4]t_2, t_2[8,10]t_3, t_1[9,16]t_3\}$. *KB* is *consistent* (i.e., there is at least a solution satisfying all the constraints), $\{t_1=0, t_2=2, t_3=10\}$ is a *solution*, and $KB'=\{t_1[2,4]t_2, t_2[8,10]t_3, t_1[10,14]t_3\}$ is the *minimal network* (notably, the inferred distances between t_1 and t_3 are *[10,14]*). ■

In STP (and in many other temporal frameworks), a set of temporal constraints can be represented by a graph, and temporal reasoning is performed by an *all-to-all shortest path algorithm* (e.g., Floyd-Warshall's one) which repeatedly applies the operations of *intersection* (∩) and *composition* (@) of constraints to *check the consistency* and produce the *minimal network* (see Fig. 1 below). In short, given a constraint between t_i and t_k, and a constraint between t_k and t_j, *composition* determines the implied constraint between t_i and t_j. Intuitively, the minimum distance between t_i and t_j is obtained by summing the minimum distance between t_i and t_k to the minimum distance between t_k and t_j (similarly for the maximum distance). On the other hand, given two constraints between t_i and t_j, *intersection* evaluates the tightest implied constraints. Intuitively, the resulting constraint is simply obtained by intersecting the two ranges of possible distances, such that both constraints must hold (if the intersection is empty, the two constraints are *inconsistent*). For instance, in the above example, the new constraint between t_1 and t_3 is evaluated as $t_1[9,16]t_3 \cap (t_1[2,4]t_2 @ t_2[8,10]t_3)$, where $t_1[2,4]t_2 @ t_2[8,10]t_3 = t_1[10,14]t_3$ and $t_1[9,16]t_3 \cap t_1[10,14]t_3 = t_1[10,14]t_3$.

```
for k:=1 to N do
  for i:=1 to N do
    for j:=1 to N do
          C(i,j) = C(i,j) ∩ (C(i,k) @ C(k,j))
```

Fig. 1. Floyd-Warshall's algorithm. *i*, *j*, and *k* denote variables (time points, in the temporal reasoning context), *C(l,m)* de-

notes the constraint between *l* and *m*, and *N* is the number of time points.

The complexity of Floyd-Warshall's algorithm is $\Theta\left(N^3 * max(T_{\cap}, T_{@})\right)$, where, N is the number of time points in the knowledge base, and $T_{\cap}$ and $T_{@}$ represent the time needed to calculate $\cap$ and @, respectively. Floyd-Warshall's algorithm has been proved to be *correct* and *complete* on STP (i.e., all the implied constraints are computed), and provides as output the minimal network of the input constraints, or reports an inconsistency (Dechter et al., 1991).

The above family of temporal reasoning approaches constitutes the background for our work, and, in the next subsection, we show how it is being extended towards the treatment of "non-crisp" temporal constraints. Before that, however, it is worth highlighting that, recently, a main research direction in the AI research aims at extending distance-based temporal constraints to cope with other forms of uncertainty. For instance, Simple Temporal Network with Uncertainty (STNU) (Morris et al., 2001) supports temporal constraints contexts in which the temporal duration of some actions is not under the control of the agent(s). Conditional Simple Temporal Network (CSTN) supports conditioned temporal constraints between time points, in which conditions can only be observed in real time (Hunsberger et al., 2015). In STNU and CSTN, a crucial temporal reasoning task is to determine whether the network of constraints is dynamically consistent, e.g., in the case of STNU, whether there exists a strategy for executing its time-points such that all relevant constraints are guaranteed to be satisfied no matter which are the durations of non controllable actions. Very recent evolutions in this area are provided, e.g., in (Cairo et al., 2018; Combi et al., 2019; Combi & Posenato, 2018; Hunsberger & Posenato, 2018; Zavatteri et al., 2019; Zavatteri & Viganò, 2019).

2.2 Moving towards "non-crisp" temporal constraints

Traditional temporal reasoning approaches are based on the Constraint Satisfaction Problem (CSP) framework. Temporal constraints are "crisp": they represent set of equally possible/preferable constraints between two time units. Unfortunately, such approaches inherit also the limitations of CSP, concerning flexibility and its limited representation of uncertainty (see, e.g., the discussion in (Dubois et al., 1996)). In many real contexts, constraints are not strict. Therefore, the AI literature has moved towards "non-crisp" constraints, to take into account *probabilities* or *preferences*. Starting from the seminal work in (Dubois et al., 1996), a large number of temporal reasoning approaches based on the fuzzy CSP have been devised.

In the following, we restrict our attention to the fuzzy extension of the constraints discussed in Section 2.1 above. Considering qualitative temporal constraints, Riabov and Trudel (Ryabov & Trudel, 2004) pair each Allen's basic interval relations with a probability. Temporal reasoning is performed through inversion, composition, and addition operations, which are extended in order to combine, besides the temporal relationships, also their probability. Similarly, Mouhoub and Liu (Mouhoub & Liu, 2008) propose an adaptation of the probabilistic CSP framework and Badaloni and Giacomin (Badaloni & Giacomin, 2006) extend Allen's interval relations with a preference degree. As regards "non-crisp" quantitative temporal constraints, in (Terenziani & Andolina, 2017) the STP framework has been extended by associating probabilities to distances. The authors define new operations of intersection and composition, which also combine the probabilities associated with the distances in the constraints. *Floyd-Warshall*'s algorithm is used in order to evaluate the minimal network of such extended constraints. On the other hand, Khatib et al. (Khatib et al., 2001) have extended the STP and the TCSP (Temporal Constraint Satisfaction Problem) frameworks (Dechter et al., 1991) to allow for reasoning about temporal preferences, to identify an optimal solution. In a recent paper, Terenziani et al. (Terenziani et al., 2017) have associated a preference with each possible distance between time points and have used an adaptation of the *Compute-Summaries* algorithm to evaluate the minimal network. Additionally, they have proposed query-answering facilities to explore the minimal network with preferences. Mouhoub and Sukpan (Mouhoub & Sukpan, 2008) deal with both preferential quantitative and qualitative temporal constraints and perform temporal reasoning to find a *solution* optimizing preferences.

Despite the huge amount of work in this area, however, no approach in the literature has faced yet the task of extending STP temporal constraints and computing the minimal network considering **both** preferences **and** probabilities (except the preliminary short AIME'19 paper (Terenziani & Andolina, 2019)).

2.3 Temporal constraints in CIG systems

Temporal constraints are an intrinsic part of many CPGs and CIGs. Therefore, they are explicitly recorded in many CPGs and CIGs. Notably, Kamisalic et al. have proposed an interesting approach to mine CPG temporal constraints from medical data, and to represent them (Kamisalic et al., 2018). This contribution is very important, since it may be used as a starting point to derive temporal constraints from experience, for those CPGs which do not natively contain them. Temporal constraints are essential in many different tasks involving CIGs (see, e.g., the survey (Terenziani et al., 2008)). For instance, several CIG execution modules take into account CIG temporal constraints in order to make recommendations about *when* CIG actions should be executed on the patient. As a consequence, temporal constraints and temporal reasoning have been faced by several different approaches in the area (see, e.g., the survey in (Terenziani et al., 2008)). To mention few examples, consider e.g. ASBRU (Duftschmid et al., 2002; Shahar et al., 1998), GLARE

(Terenziani et al., 2001), and (Combi et al., 2015). ASBRU provides temporal annotations to express the minimum and maximum starting time and ending time of durative actions (intervals), as well as their minimal and maximal duration. Also, cyclical annotations are supported, to cope with periodically repeated actions. Also, GLARE provides a rich temporal constraint language, which extends the STP framework (Dechter et al., 1991) (see also Section 2.1) to consider also periodic constraints. GLARE provides temporal reasoning algorithms to propagate such constraints, and proves their correctness and completeness (Anselma et al., 2006). In (Combi et al., 2015), the authors propose a conceptual model for CIGs which takes into account also temporal constraints and temporal constraint propagation, considering also uncertainty and alternative paths.

In the last decade, several CIG approaches have started to be extended to support the treatment of ***comorbidities***. The treatment of temporal constraints and of temporal reasoning is fundamental in such a context, and, in particular, for the detection of temporal interactions that can actually occur in time (see the discussion in the introduction).

Indeed, knowledge about the effects of CIGs' actions and about the possible interplay between such effects may be exploited in order to detect the possibility of interactions between CIGs (see, e.g., (Zamborlini et al., 2017)). However, only a temporal analysis concerning when actions have to be executed and when their effects can occur can discriminate between those interactions that may effectively occur in time, and the ones that cannot. Indeed, interactions (between the effects of CIG actions) may occur only in case effects may *intersect in time*. Therefore, detecting *temporally possible interactions* is a challenging task, requiring temporal knowledge (in the form of *temporal constraints* between CIG actions, and between actions and their effects) and *temporal reasoning* (e.g., in the form of *temporal constraint propagation*). Some of the current AIM approaches in the literature face such a problem, taking advantage of general frameworks, that natively support representation of and reasoning about constraints, including temporal ones. Five significant and recent examples are the approaches in (Jafarpour et al., 2019; Michalowski et al., 2021; Piovesan et al., 2020; Van Woensel et al., 2021; Wilk et al., 2017). (Wilk et al., 2017) proposes a framework based on *First Order Logic* (*FOL*) for identifying and managing interactions between multiple CIGs. Different predicates are defined in the logic in order to represent temporal aspects (consider, e.g., *duration, startTime, overlap, endTime*). Reasoning about constraints is directly performed by the theorem prover for the logic. Similarly, also the recent approach by Jafarpour et al. (Jafarpour et al., 2019) uses an FOL (plus OWL2) encoding to model execution-time dynamic integration of CIGs. Temporal constraints regarding minimum and maximum delay, and duration of CIGs' tasks are explicitly modelled as FOL predicates, as well as the maximal durations of the effects\results of the tasks. Such constraints are used, e.g., in the definition of the policies to manage interactions (e.g., *SimulTasksPolicy, DelayingTaskPolicy*). Van Woensel et al. (Van Woensel et al., 2021) basically extends the previous works and exploits a dynamic (continuous) planning module coping with the temporal aspects of CIG integration. On the other hand, in (Michalowski et al., 2021), the problem of interaction mitigation in multiple CIGs is modeled as a planning problem, using PDDL 2.1, which provides support for durative actions, and different forms of temporal constraints (e.g., delays) (Fox & Long, 2003). The Optic planner is used to find solutions to the planning problem, considering such constraints (Coles et al., 2010). Interestingly, such an approach considers also patient's preferences to merge CIGs. Finally, (Piovesan et al., 2020) proposes an Answer Set Programming encoding of the problem of analysing a-posteriori the conformance of the treatment of a comorbid patient (encoded into the patient's log) with respect to the CIGs' recommendations. Answer Set Programming is thus used to model, besides the other pieces of knowledge, also the temporal constraints between CIGs' actions, and the delays and durations of the effects of such actions, and to reason with them. Other works in literature leave the management of temporal aspects as future work. Among them, the work a work of particular interest is the one in (Kogan et al., 2020), based on the well-known framework ProForma and adopting a goal-based methodology for CIG combination.

On the other hand, the GLARE project (Anselma et al., 2017; Piovesan et al., 2018; Terenziani et al., 2001) follows the mainstream of AI research aiming at proposing a temporal framework specialized for the representation and the reasoning about temporal constraints. Specifically, (Anselma et al., 2017) proposes a temporal reasoning framework based on the well-known STP framework (Dechter et al., 1991), and provides physicians with a direct support to face many different temporal reasoning tasks involved in the detection and management of interactions. In GLARE-SSCPM's architecture, such a framework acts as a *temporal knowledge-server*, which interacts with several other modules in order to support physicians in the detection and management of CIGs' interactions (Piovesan et al., 2018). Such issues will be further discussed in Section 3 below.

3 GLARE AND GLARE-SSCPM

GLARE (GuideLine Acquisition Representation and Execution, (Terenziani et al., 2001)) is a domain-independent framework supporting physicians in the acquisition and execution of CIGs. It has been developed in a collaboration between the University of Eastern Piedmont and the San Giovanni hospital in Turin (one of the major hospitals in Italy), that have started in 1997. Among the other features, GLARE provides a user-friendly graphical user interface

(GUI) and a representation formalism supporting physicians both in the acquisition of CIGs and a tool that takes into account the patient's electronic clinical record, and supports physicians in the execution of a CIG on a specific patient. In the formalism of GLARE, CIGs are modelled as conditional and hierarchical graphs, whose nodes represent actions or decisions and whose arcs model the control flow relations and the temporal constraints between them (see examples in the following). GLARE supports the distinction between *atomic actions* (simple steps in a CIG) and *composite actions* (plans). Atomic actions can be: *work actions* (a procedure which must be executed), *pharmacological actions* (a drug to be administered), *query actions* (retrieval of information from the clinical record/examinations) or *decisions* (choice among different alternatives).

Fig. 2 shows an example of a CIG acquired using GLARE. The reported CIG is a simplified version of the guideline for the treatment of Urinary Tract Infections (UTI) provided by the British National Institute for Health and Care Excellence (NICE; (*NICE | The National Institute for Health and Care Excellence*, n.d.)). The figure shows most of the GLARE elements. Yellow diamonds represent decisions. For instance, the decision "CATHETER" distinguishes between the treatment of patients suffering from UTI with or without a urinary catheter. Blue circles represent work actions. For instance, "Hospitalize patient" recommends hospitalizing patients having symptoms (e.g. sepsis) suggesting a more serious illness or condition. In such a case the current CIG ends because this guideline does not include the treatment of hospitalized patients. Finally, red hexagons represent composite actions, that can be further expanded in term of their components. For instance, "UTI lifestyle" is a composite action for the follow-up of patients treated for UTI, containing a plan with recommendations (such as drinking enough fluids - to avoid dehydration - and self-care medications) to prevent relapses. In the figure we have already expanded the composite action for the treatment of the upper UTI (which includes the part of the graph starting with the decision "SERIOUS SYMPTOMS?"). On the other hand, the composite action "Nalidixic Acid treatment" has not been expanded in the figure. It contains a repetition of the pharmacological action "Nalidixic Acid Administration", to be performed periodically for 1 or two weeks with a delay between two consecutive actions ranging from 5 to 7 hours (i.e., 20 to 28 time units – see Ex.1).

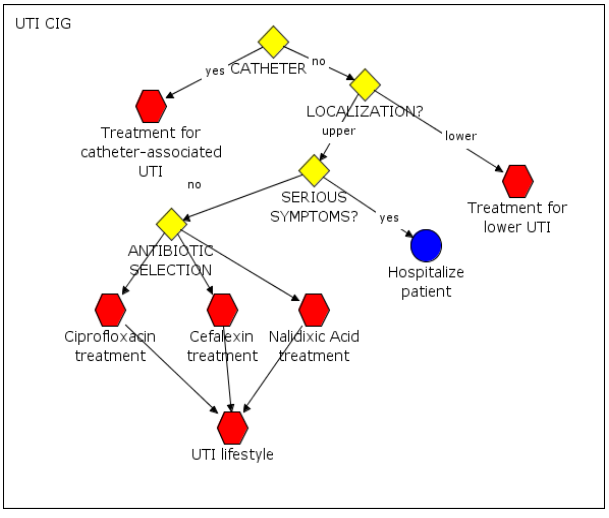

Fig. 2. UTI CIG modeled with the GLARE formalism.

GLARE's architecture is modular and open. Along the years, new modules and\or methodologies have been added to deal with CIG contextualization (Terenziani et al., 2004), temporal reasoning (Anselma et al., 2006), cost-benefit analysis (Montani & Terenziani, 2006), model-based verification (Bottrighi et al., 2010), conformance analysis (Spiotta et al., 2017) and agent delegation (Bottrighi et al., 2019).

Additionally, GLARE has also been extended with a *Knowledge Manager*, to cope with additional CIG-independent medical knowledge (Piovesan et al., 2014). The Knowledge Manager adopts an OWL ontological model (built using Protégé; (Musen, 2015)), which also integrates parts of the SNOMED CT (International Health Terminology Standards Development Organisation, 2015) and ACT (WHO Collaborating Centre for Drug Statistics Methodology, n.d.) ontologies. GLARE's actions can be linked to ontological elements. The ontological model enriches GLARE with additional basic medical knowledge, associating actions with their effects and intentions. Effect and intentions are modeled as variations of the patient's status attributes. Moreover, the ontological model represents the relations between effects/intentions, such as causal relations or interactions. In (Anselma et al., 2017) the ontological model has been enriched with temporal information. In particular, the properties relating the actions/drugs to their effects have been

extended to contain the delays between the action execution/drug administration and the onset and the end of the effects. Notably, GLARE' Knowledge Manager module is paired with an OWL reasoners providing inferences about the previous elements.

Fig. 3 (solid lines) shows part of the ontological knowledge of GLARE for the actions "Calcium Carbonate Administration" and "Nalidixic Acid Administration" involved in Ex.1. In particular, the ontological model associates both the actions with the drugs they prescribe (property "substance") and each drug is associated with its effects (property "hasEffects"). For instance, the action "Calcium Carbonate Administration" is related to the substance "Calcium Carbonate", that has the effect "Decrease Gastric Acidity". Such an effect, however, causes also a decreasing of the gastric absorption capability ("Decrease Gastric Absorption"). Following such a model, the *Knowledge Manager* of GLARE is able to infer that each CIG action linked to the concept "Calcium Carbonate Administration" has, as effects, the decreasing of gastric acidity and the decreasing of the gastric absorption. Such a knowledge can be exploited, for instance, to study the interactions between CIG actions for comorbid patients (see below).

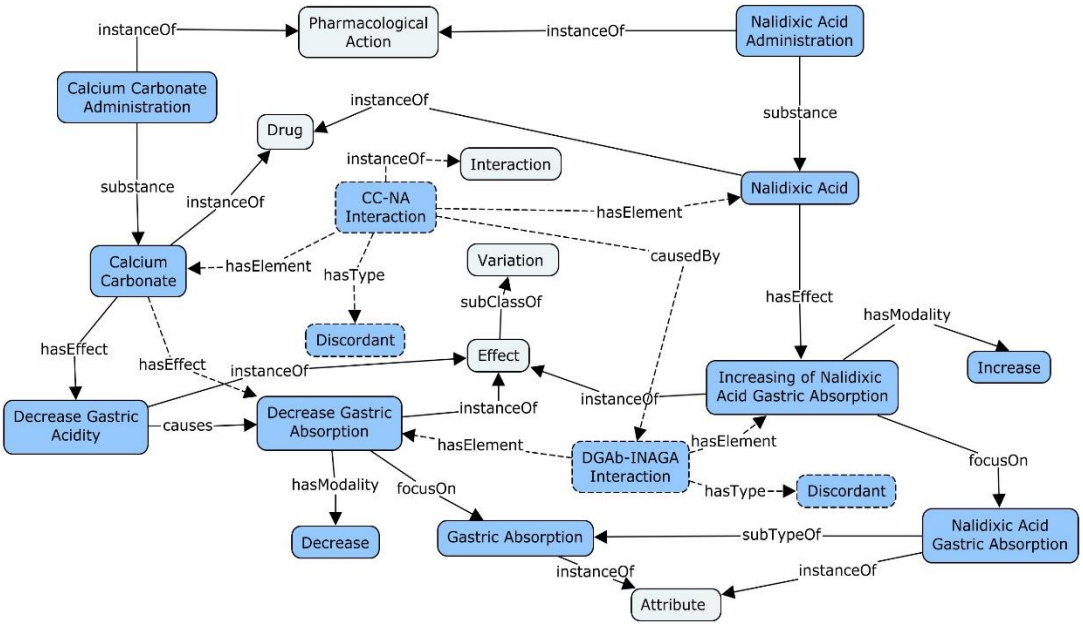

Fig. 3. Part of the ontological model (solid lines) of GLARE exploited to infer the interaction between the actions of Ex.1. Dashed nodes and arcs represent the inferences of the reasoner.

GLARE-SSCPM (Piovesan et al., 2018) is the GLARE extension devoted to the treatment of comorbidities. GLARE-SSCPM faces all the three "core" tasks in the CIG-based treatment of comorbid patients: (i) the detection of interactions occurring between CIGs (Piovesan et al., 2014, 2015), (ii) the management of the interactions (Piovesan & Terenziani, 2015), and (iii) the final merging of (parts of the) CIGs (Piovesan & Terenziani, 2016). With respect to the other related approaches in the literature, GLARE-SSCPM is characterized by a high degree of interaction with user-physicians, to interactively support, "step-by-step", their decisions, and by a specific attention to the temporal dimension. In particular, GLARE-SSCPM supports physicians by adopting the "***focus***, ***hypothesize and test***" methodology. In a session of work using GLARE-SSCPM, a physician can (i) take advantage of GLARE-SSCPM to ***focus*** on the parts of the CIGs that are relevant to the current status of the patient, and ask it to automatically ***detect*** the possible interactions between the focused actions. Then (ii) GLARE-SSCPM provides user-physicians with different options for ***managing*** an interaction, giving them a way of simulating the application of each option, until one of them is chosen by the physicians. Finally, (iii) in case multiple interactions have been managed, GLARE-SSCPM automatically *checks* the general consistency of the managements. In the case of consistency, it provides a "***merge***" of the (focused parts of the) CIGs, while, in the case of inconsistency, physicians may check alternative managing options (following the "hypothesize and test" methodology).

The general architecture and behaviour of GLARE-SSCPM is described in (Piovesan et al., 2018), while the specific treatment of focusing, automatic detection, interaction management and merge is described in (Piovesan et al., 2014; Piovesan & Terenziani, 2015, 2016), respectively. In the following, we just mention focusing, management and merge, and then we briefly describe interaction detection and temporal reasoning, which concern the focus of this paper.

Focusing. Since CIGs may consists of hundreds of actions, and (in GLARE) can be structured at different levels of abstraction, GLARE-SSCPM provides physicians with a facility, to support them in the tasks of focusing on the parts of the CIGs which are currently "relevant" for the patient at hand, and of choosing the appropriate level of abstraction. Such a tool provides a graphical interface, and supports physicians in the "navigation" of the CIGs, at different levels of abstraction. A "navigation tree" is maintained, to provide physicians also a way to backtrack to previous focuses.

Interaction Management. Analysing the medical literature, it emerges that different options are used by physicians to manage (i.e., "solve") interactions, and the choice between them is usually context and patient-dependent. Therefore, GLARE-SSCPM provides physicians with 8 different management options. Management options leads to local changes to the CIGs, to avoid undesirable interactions and to enforce desirable ones. For example, the "safe alternative" option allows physicians to avoid an interaction by choosing alternative paths in the CIGs, while the "replanning" option substitute an interacting action with an automatically determined plan achieving the same intentions of the interacting action. Interestingly, in GLARE-SSCPM, all the 8 management options are achieved on top of three basic reasoning techniques: *Backward Navigation* on the CIGs, *Goal Based Planning* (taking advantage of the Knowledge Manager and of its Knowledge Base), and Temporal Reasoning.

CIG conciliation. In case more than one management is needed (due to multiple interactions to be managed), their global consistency must be checked, and a "local merge" of the involved parts of the CIG must be provided. Such a task is automatically managed by GLARE-SSCPM taking advantage of the CSP (Constraint Satisfaction Problem) framework.

Interaction detection. Given a set S_A of actions from different CIGs (in case no focusing is performed, all the actions in the CIGs – used for the given patient – are considered), GLARE-SSCPM automatically detects the possible interactions between them. Such a detection is achieved in two steps. In the first step, GLARE-SSCPM exploits the ontological knowledge, a set of Semantic Web Rules (SWRLs) and the OWL reasoner to retrieve all the possible interactions between the intentions, effects and (in case of pharmacological actions) prescribed drugs of the actions in S_A. As an example, the part of the ontological model shown in Fig. 3 can be used to detect an interaction between the actions CCA and NAA of Ex.1 (in Ex.1, we have S_A = {"Nalidixic Acid Gastric Administration", "Calcium Carbonate Administration"}). Notably, dashed elements in Fig. 3 represent the *inferences of the reasoner* .

Given S_A, the reasoner focuses on the effects, the intentions and the drugs of the actions in S_A to detect possible interactions. In our example, "Nalidixic Acid Gastric Administration" is an administration of the substance (*substance* relationship) "Nalidixic Acid" which has as effect (*hasEffect* relationship) "Increasing of Nalidixic Acid Gastric Absorption"; "Calcium Carbonate Administration" is an aministration of the substance "Calcium Carbonate", which has as effect "Decrease Gastric Acidity".

Exploiting the causal relationship between the effects "Decrease Gastric Acidity" and "Decrease Gastric Absorption", the reasoner infers that also "Decrease Gastric Absorption" is an effect of the drug Calcium Carbonate (dashed edge "hasEffect"). "Increasing of Nalidixic Acid Gastric Absorption" focuses on (*focusOn* relationship) the attribute "Nalidixic Acid Gastric Absorption", that is a sub-type (*subTypeOf* relationship) of the attribute "Gastric Absorption". Notably, such an attribute is also the focus of "Decrease Gastric Absorption". As a consequence, an interaction between such two effects is inferred (see the dashed concept "DGAb-INAGA Interaction" in Fig. 3). Moreover, "Increasing of Nalidixic Acid Gastric Absorption" causes an increase (*hasModality* arc stemming from "Increasing of Nalidixic Acid Gastric Absorption"), while "Decrease Gastric Absorption" causes a decrease of "Gastric Absorption". Therefore, through the following SWRL rule, a discordance is detected between the two effects.

```
Interaction(?i) ^ hasElement(?i, ?v1) ^ hasElement(?i, ?v2) ^ focusOn(?v1, ?a1) ^ fo-
cusOn(?v2, ?a2) ^ subTypeOf(?a2, ?a1) ^ hasModality(?v1, ?m1) ^ hasModality(?v2, ?m2)
^ differentFrom(?m1, ?m2) -> hasType(?i, Discordant)
```

Finally, since the interaction detected at the previous step involves two effects of the drugs "Nalidixic Acid" and "Calcium Carbonate", a discordance interaction ("CC-NA Interaction" in Fig. 3) is inferred between them through the following SWRL rule:

```
Interaction(?i) ^ hasElement(?i, ?d1) ^ hasElement(?i, ?d2) ^ Drug(?d1) ^ Drug(?d2) ^
Interaction(?i1) ^ hasElement(?i1, ?v1) ^ hasElement(?i1, ?v2) ^ hasEffect(?d1, ?v1) ^
hasEffect(?d2, ?v2) ^ hasType(?i1, ?t1) -> hasType(?i, ?t1) ^ causedBy(?i, ?i1)
```

As a consequence, the interaction "CC-NA interaction" between the concepts "Calcium Carbonate Administration" and "Nalidixic Acid Administration" is inferred and reported to the user physician. If needed, the explanations provided by the OWL reasoner are also shown to the user physicians by the GUI of the Interaction Detection module of GLARE-SSCPM.

The interactions detected at this step, without considering time, are only hypothetical. Indeed, if the interacting effects do not overlap in time, the interaction does not actually occur. For such a reason, GLARE-SSCPM adopt a temporal reasoner to check whether the detected interactions can temporally occur, or not.

Temporal reasoning. In the second step, for each detected interaction, temporal reasoning is used to check whether it can actually occur or not, given the temporal information (i) in the CIGs (in the form of temporal constraints between –the endpoints of– CIG actions – e.g., delays between action), (ii) in the (temporally extended) ontological model, concerning the delay and duration of the effects of actions, and (iii) the time of execution of previous actions on the patient. Following the AI stream of research described in Section 2, in GLARE-SSCPM we have developed a specialised temporal reasoner to manage such temporal pieces of information, and to reason (in the form of constraint propagation) about them (Anselma et al., 2017). Moreover, we enriched the knowledge model of GLARE to represent temporal constraints between actions and/or effects. The temporal reasoner acts as a knowledge server, to which temporal problems may be delegated (notably, also interaction management takes advantage of the temporal reasoner, as described in (Piovesan et al., 2018)).
The temporal approach we developed in (Anselma et al., 2017) provides a wide set of facilities which are specific of the CIG application domain. However, the "kernel" of the approach is the "classical" STP framework discussed in Section 2.1. A specific interface module, composed by three submodules, is used in order to retrieve the temporal constraints (i)-(iii), to map them into STP constraints, and to give them as input to the STP temporal reasoner. Specifically, constraints (i) are retrieved through a navigation into the CIGs (which focus only, in each CIG, on the path in the CIG which is being applied to the given patient); constraints (ii) are retrieved through navigation and search in the knowledge base, by considering as entries the link (to the knowledge base) contained in the actions in the CIG; constraints (iii) are simply taken from the "log" of the actions executed on the patient (which contains the starting and ending time for action execution).

Let us consider, as an example, the temporal analysis of the interaction between CCA and NAA. In GLARE-SSCPM, interaction detection (and the related temporal analysis) is performed whenever required by the physicians. As a consequence, in order to propose a running example, we have to fix a specific situation in which the temporal analysis is invoked (see Ex.1' below, which contextualizes the patient-independent example discussed in Ex.1). Notably, in STP, absolute times are expressed as distances (delays) from a given reference point. For the sake of simplicity, in this example we take as reference time 00:00 of the current day (X_0 henceforth), and we consider units of 15 minutes as temporal granularity. For instance, time unit 8 represents time 02:00 of the current day.

Ex.1'. We consider a comorbid patient affected by GERD and UTI. Specifically, as regards UTI, we suppose that several actions in the CIG have been already executed on the patient: the decisions "CATHETER" (chosen branch: "no"), "LOCALIZATION" (chosen branch: "upper"; i.e., the patient has upper UTI), "SERIOUS SYMPTOMS?" (chosen branch: "no") and "ANTIBIOTIC SELECTION" (chosen branch: "Nalidixic Acid treatment"). We also suppose that the patient has already started the NAA therapy, and that today the patient has already had an administration (node NAA0 in Fig. 4) at unit 28.

As regards the GERD CIG, we suppose that today the RS decision is executed at the time unit 52, with a positive answer, so that CCA has to be administered to the patient.

First, GLARE-SSCPM extracts the constraints from the CIGs, the knowledge model and the log. Notably, the temporal constraint that the ending time of all the durative elements (e.g., durative actions, effects) is equal or after their starting time is automatically inserted by the framework. A detailed description of the automatic extraction procedure is provided in (Anselma et al., 2017). Fig. 4 shows the STP derived from the constraints extracted to temporally analyze the interaction in Ex.1+Ex.1' above. Specifically,

- the constraints X_0 [28,28] NAA0 and X_0 [52,52] RS are extracted from the log (storing the time when actions are executed on the patient; see Ex.1'),
- the constraints NAA0 [20,28] NAA and RS [0,6] CCA are extracted from the UTI and GERD CIGs respectively
- the constraints between NAA and the starting ($INAGA_s$) and ending ($INAGA_e$) times of its effect Increase of Nalidixic Acid Gastric Absorption, and between CCA and the starting ($DGAb_s$) and ending ($DGAb_e$) times of its effect Decrease of Gastric Absorption are extracted from the ontological model.

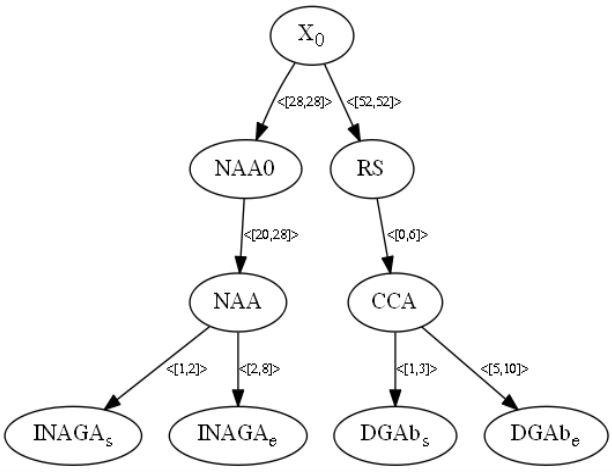

Fig. 4. STP extracted for the analysis of the interaction between CCA and NAA of Ex.1.

The temporal reasoner of GLARE-SSCPM is based on the STP framework, and propagates the input constraints to provide the minimal network (see Section 2.1). For example, some of the constraints in the minimal network are shown in Fig.5 below. Since the minimal network is not easy to interpret for non-expert users, GLARE-SSCPM provides several facilities, including a rich language to query the minimal network. In particular, GLARE-SSCPM's "Interaction" facility allows physicians to check the possible temporal occurrence of an interaction, by checking whether

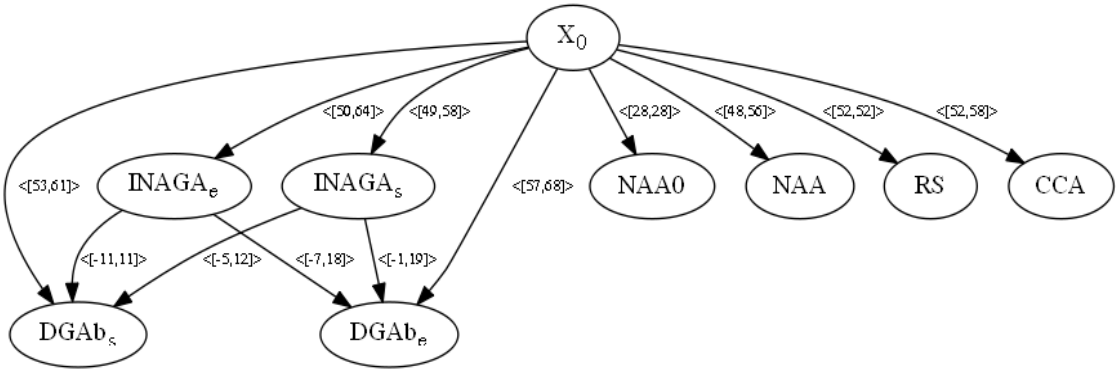

Fig. 5. Part of the minimal network for Ex.1, representing the temporal constraints between the actions/effects and the reference time X_0, and the temporal constraints between the interacting effects.

the interacting effects must, may or cannot intersect in time. Additionally, to facilitate users in the analysis of temporal constraints, GLARE-SSCPM provides a graphical interface to visualize them (see (Anselma et al., 2017) for a detailed description)In general, the graphical facility allows users to specify a set of time points, and provides as output a graphical representation of their temporal location, with respect to the timeline (i.e. their distance with respect to the reference time X_0). Specifically, such a general facility can be used in order to study interactions, e.g., by visualizing the temporal location of the endpoints of the effects of (potentially interacting) actions. For example, Fig. 6 shows the GUI for the analysis of the interaction between the effects DGAb and INAGA of CCA and NAA in Ex.1+Ex.1'. The GUI shows in a timeline (starting at the chosen X_0 – that in our example is the time 00:00 of the current day) the time intervals (represented as rectangles) when the interacting effects (DGAb and INAGA) can start and end. For instance, the top rectangle in the figure represents the time when DGAb can start. Through this representation, it is easier to notice that the two effects may overlap in time, but they can also be disjoint.

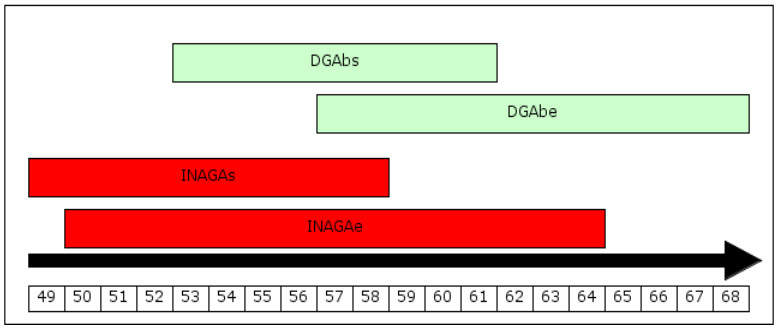

Fig. 6. GUI for the analysis of the interaction of Ex.1. Coloured rectangles represent the times in which the interacting effects can start/end expressed as delays from the reference time X_0.

GLARE-SSCPM also supports Boolean queries. For instance, users can ask whether, given the minimal network, DGAb and NAGA intersect. The answer, in our example, would be "possibly"[2]. Finally, GLARE-SSCPM also supports hypothetical queries, i.e., queries to be answered in a context in which a set of STP constraints is hypothesized (see the discussion in Section 6.5).

4 Probabilistic+Preferential Quantitative Temporal Networks

We aim at extending STP to support the possibility to associate *both* preferences *and* probabilities to the possible distances between time points. In our approach, time points may represent:

(i) the time of occurrence of "instantaneous" events, or

(ii) the endpoints of the time intervals in which durative events occur.

For instance, in our encoding of Ex.1+Ex.1′ (see Ex.3 below) we suppose, with no loss of generality, that Calcium Carbonate Administration (CCA) and Nalidixic Acid Administration (NAA) are instantaneous events. Of course, their effects "Decreasing Gastric Absorption" (DGAb) and "Increase of Nalidixic Acid Gastric Absorption" (INAGA) are durative.

Though our temporal reasoning approach focuses on time points, we maintain two tables, encoding the correspondence between instantaneous events and their time of occurrence, and durative events and the starting and ending time of their interval of occurrence.

To model preferences and probabilities along paths of events, we need to consider (i) "purely" preferential constraints, (ii) "purely" probabilistic constraints, and (iii) "mixed" probabilistic+preferential constraints. We represent the above constraints homogeneously, where each constraint has both a probability and a preference, and we use two special symbols, namely "%" and "", to denote the undefined probability (for type (i) constraints) and the undefined preference (for type (ii) constraints), respectively. When probabilities are undefined (value "%"), we adopt the usual interpretation that all the alternatives have the same probability. Without loss of generality, we assume that both probability and preferences are defined over the domain $[0,1] \in R$. Thus, we define the domains *PB* (for probabilities) and *PF* (for preferences) as follows.

Terminology. We term ***PB*** the domain $[0,1] \cup \{\%\}$ of probabilities, and ***PF*** the domain $[0,1] \cup \{\#\}$ of preferences. ■

Definition.

- let $t_i, t_j \in R$ be time points,
- let $p_1, \dots, p_n \in R$ be probabilities; $0 < p_1 \leq 1, \dots, 0 < p_n \leq 1$ or $p_1 = \dots = p_n = \%$,
- let $P_1, \dots, P_n \in R$ be preferences; $0 \leq P_1 \leq 1, \dots, 0 \leq P_n \leq 1$ or $P_1 = \dots = P_n = \#$,

[2]It is worth noticing that, though "crisp" STP constraints do not model preferences nor probabilities, they may still be indeterminate (i.e., not exact). As a consequence, in the "crisp" case, the answers to Boolean queries are not just yes/no, but yes/possibly/no. As an example, consider two time points t_1 and t_2, and a query asking whether t_2 follows t_1 (i.e., $t_2 > t_1$). The answer to such a query would be "yes" in case, e.g. the STP constraint between t_1 and t_2 is $t_1[1,2]t_2$, "possibly" in case the constraint is $t_1[-1,+1]t_2$, "no" in case the constraint is $t_1[-2,-1]t_2$.

- let $d_1, \dots, d_n \in Z$ be distances (between points).

A **Probabilistic+Preferential Quantitative Temporal Label** (P+PQTL) is a list $\langle (d_1, p_1, P_1), \dots, (d_n, p_n, P_n) \rangle$ where either $p_1 = \dots = p_n = \%$ or $p_1 + \dots + p_n = 1$. In the following, we denote by L the domain of such labels.

A **Probabilistic+Preferential Quantitative Temporal Constraint** (P+PQTC) is a constraint of the form $t_i C t_j$ where $C \in L$ and $t_i, t_j \in R$ are time points.

A **Probabilistic+Preferential Quantitative Temporal Network** (P+PQTN) is a directed graph $G = \langle V, E \rangle$ with an edge labelling λ, where V is a set $\{t_1, \dots, t_{|V|}\}$ of time points, $E \subseteq V \times V$, and $\lambda: E \to L$. ■

The intended meaning of a constraint $t_i \langle (d_1, p_1, P_1), \dots, (d_n, p_n, P_n) \rangle t_j$ is that the distance $t_j - t_i$ between t_j and t_i is d_1 with probability p_1 and preference P_1, or … or d_n with probability p_n and preference P_n.

Ex.3. For example, the P+PQTC

CCA <(1,0.4,#), (2,0.4,#), (3,0.2,#)> DGAb_s

represents the constraint between CCA and the starting point of DGAb (denoted as DGAb_s) in Ex.1+Ex.1′, and

RS <(0,%,1),(1,%,1), (2,%,0.75), (3,%,0.75), (4,%,0.5), (5,%,0.5), (6,%,0.25)> CCA.

represents the temporal constraint in Ex.1+Ex.1′ relating the decision RS to the action CCA (in the CIG for GERD). ■

The P+PQTN associated with Ex.1+Ex.1′ is shown in Fig. 7.

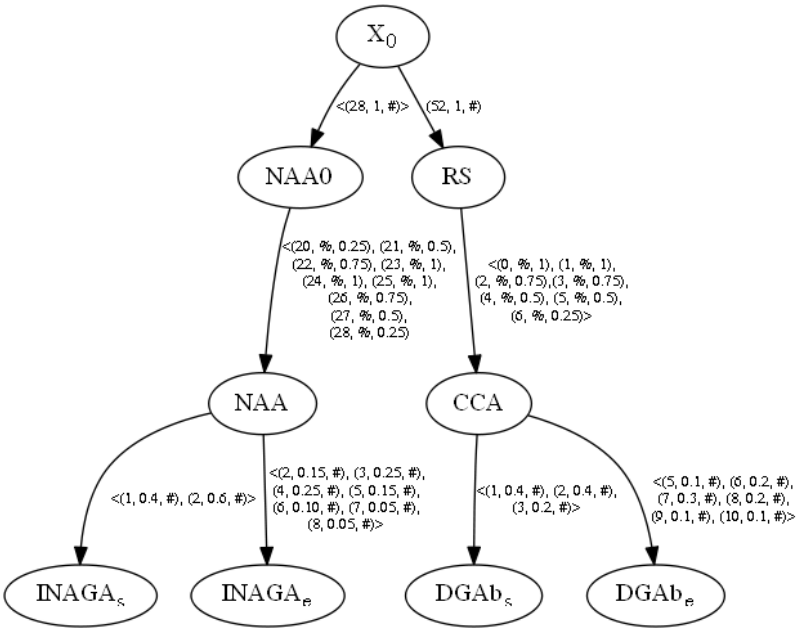

Fig. 7 Graphical representation of the P+PQTN for Ex.1+Ex.1′.

5 TEMPORAL REASONING ON P+PQTNS

As discussed in Section 4 above, we extend STP (with discrete distances) to include both probabilities *and* preferences. As in STP, we use Floyd-Warshall's algorithm to perform temporal reasoning. However, we have to extend it to apply to P+PQTNs. To achieve such a goal, we extend the operators of intersection and composition to operate also on *both* probabilities and preferences. In the following, we denote by $\cap_{pP}$ and $@_{pP}$ our extended intersection and composition operators.

5.1 Intersection and composition operators

Concerning distances, both $\cap_{pP}$ and $@_{pP}$ work as the corresponding STP operators.

The ***intersection*** operator $\cap_{pP}$ "merges" two constraints $C_1 = \langle (d'_1, p_{d'_1}, P_{d'_1}), \dots, (d'_n, p_{d'_n}, P_{d'_n}) \rangle$ and $C_2 =$

$\langle(d''_1, p_{d''_1}, P_{d''_1}), \dots, (d''_m, p_{d''_m}, P_{d''_m})\rangle$ regarding the same pair of time points. As in STP, the output distances are obtained as the intersection of the input distances. As regard probabilities, as in the probabilistic approaches in the literature, we consider each constraint as independent of the others. The intersection operator evaluates those distance values that are in common with both input constraints, and, given the independence, it multiplies the corresponding probabilities. Intersection computes all and only the distance values that are possible between the two input points. From the probabilistic point of view, such distances form a new exhaustive set of basic events, and therefore their probabilities have to be normalized.

Considering defined preferences (i.e., preferences different from ""), we adopt the usual approach that, for each distance in the output, its preference is the *minimum* of the preferences in the input constraints. On the other hand, the intersection of a defined preference P with an undefined one is the defined preference P, while the intersection of two undefined preferences is the undefined preference. The formal definition is the following:

Definition. Intersection ($\cap_{pP}$). Given two P+PTQLs

$C_1 = \langle(d'_1, p_{d'_1}, P_{d'_1}), \dots, (d'_n, p_{d'_n}, P_{d'_n})\rangle, C_2 = \langle(d''_1, p_{d''_1}, P_{d''_1}), \dots, (d''_m, p_{d''_m}, P_{d''_m})\rangle$relating two time points, their intersection C is defined as follows. Let $\{d_1, \dots, d_k\} = \{d'_1, \dots, d'_n\} \cap \{d''_1, \dots, d''_m\}$, and let $p_X(d)$ and $P_X(d)$ be the probability and the preference of the distance d in the constraint X, then

$$C = C_1 \cap_{pP} C_2 = \langle(d_1, p_C(d_1), P_C(d_1)), \dots, (d_k, p_C(d_k), P_C(d_k))\rangle$$

where for each $d \in \{d_1, \dots, d_k\}$ the output probability $p_C(d)$ is defined as follows:

(i) if $p_{C_1}(d) = \%$ and $p_{C_2}(d) = \%$ then $p_C(d) = 1/k$,
(ii) if $p_{C_1}(d) \neq \%$ and $p_{C_2}(d) = \%$ then $p_C(d) = p_{C_1}(d)/\left(p_{C_1}(d_1) + \dots + p_{C_1}(d_k)\right)$,
(iii) if $p_{C_1}(d) = \%$ and $p_{C_2}(d) \neq \%$ then $p_C(d) = p_{C_2}(d)/\left(p_{C_2}(d_1) + \dots + p_{C_2}(d_k)\right)$,
(iv) if $p_{C_1}(d) \neq \%$ and $p_{C_2}(d) \neq \%$ then $p_C(d) = \left(p_{C_1}(d) * p_{C_2}(d)\right)/\left(p_{C_1}(d_1) * p_{C_2}(d_1) + \dots + p_{C_1}(d_k) * p_{C_2}(d_k)\right)$,
and the output preference $P_C(d)$ is defined as follows:
(i) if $P_{C_1}(d) = \#$ and $P_{C_2}(d) = \#$ then $P_C(d) = \#$,
(ii) if $P_{C_1}(d) \neq \#$ and $P_{C_2}(d) = \#$ then $P_C(d) = P_{C_1}(d)$,
(iii) if $P_{C_1}(d) = \#$ and $P_{C_2}(d) \neq \#$ then $P_C(d) = P_{C_2}(d)$,
(iv) if $P_{C_1}(d) \neq \#$ and $P_{C_2}(d) \neq \#$ then $P_C(d) = min\left(P_{C_1}(d), P_{C_2}(d)\right)$. ■
Notice that the intersection $\cap_{pP}$ may be empty if the intersection between $\{d'_1, \dots, d'_n\}$ and $\{d''_1, \dots, d''_m\}$ is empty.

Ex.4. As an example, the intersection between X <(1,0.5,1), (2,0.3,0.5), (3,0.2,0.2)> Y and X <(2,0.4,1), (3,0.4,0.8), (4,0.2,0.3)> Y is X <(2,0.6,0.5), (3,0.4,0.2)> Y. ■

Given a constraint C_1 between t_i and t_k and a constraint C_2 between t_k and t_j, the ***composition*** operator $@_{pP}$ evaluates the constraint between t_i and t_j. As in STP, the output distances are obtained as the pairwise sums of the input distances, considering all the possible combinations, that are assumed to be independent. Therefore, for each given combination of distances, we must multiply the corresponding probabilities (if they are definite). The set of all the combinations is mutually exclusive. As a consequence, the probability of each output distance is the sum of the probabilities of all the combinations generating it. When probabilities are undefined (value "%"), we adopt the usual interpretation that all the alternatives have the same probability.

As regards (definite) preferences, for each given combination of distances we minimize the corresponding preferences (i.e., we impose that the preference of a path is the minimum preference of its components), and then, for each output distance, we maximize the preference of each pair leading to such a distance (i.e., we select the preference of the "best" path). The composition of a defined preference P with an undefined one # is the defined preference P, while the composition of two undefined preferences is the undefined preference. More formally:

Definition. Composition ($@_{pP}$). Given two P+PTQLs

$$C_1 = \langle(d'_1, p_{d'_1}, P_{d'_1}), \dots, (d'_n, p_{d'_n}, P_{d'_n})\rangle, C_2 = \langle(d''_1, p_{d''_1}, P_{d''_1}), \dots, (d''_m, p_{d''_m}, P_{d''_m})\rangle$$

their composition C is defined as follows. Let $D' = \{d'_1, \dots, d'_n\}$, $D'' = \{d''_1, \dots, d''_m\}$, and $D = \{d_1, \dots, d_r\} = \{d: d = d'_i + d''_j, d'_i \in D', d''_j \in D''\}$, let $p_X(d)$ and $P_X(d)$ represent the probability and the preference of the distance d in the constraint X, respectively, and let $p_X = \%$ ($P_X = \#$) represent the fact that all distances in X have probability equal to % (preference equal to #), then

$$C = C_1 @_{pP} C_2 = \langle (d_1, p_C(d_1), P_C(d_1)), \dots, (d_r, p_C(d_r), P_C(d_r)) \rangle$$

where for each $d \in \{d_1, \dots, d_r\}$ the output probability $p_C(d)$ is defined as follows:
(i) if $p_{C_1} =$ % and $p_{C_2} =$ % then $p_C(d) = \sum_{\{\langle d', d'' \rangle : d' \in D', d'' \in D'', d'+d''=d\}} (1/n) * (1/m)$,
(ii) if $p_{C_1} \neq$ % and $p_{C_2} =$ % then $p_C(d) = \sum_{\{\langle d', d'' \rangle : d' \in D', d'' \in D'', d'+d''=d\}} p_{C_1}(d') * (1/m)$,
(iii) if $p_{C_1} =$ % and $p_{C_2} \neq$ % then $p_C(d) = \sum_{\{\langle d', d'' \rangle : d' \in D', d'' \in D'', d'+d''=d\}} p_{C_2}(d'') * (1/n)$,
(iv) if $p_{C_1} \neq$ % and $p_{C_2} \neq$ % then $p_C(d) = \sum_{\{\langle d', d'' \rangle : d' \in D', d'' \in D'', d'+d''=d\}} p_{C_1}(d') * p_{C_2}(d'')$,
and the output preference $P_C(d)$ is defined as follows:
(i) if $P_{C_1} =$ # and $P_{C_2} =$ # then $P_C(d) =$ #,
(ii) if $P_{C_1} \neq$ # and $P_{C_2} =$ # then $P_C(d) = max(\{P_{C_1}(d') : d' \in D', d'' \in D'', d' + d'' = d\})$,
(iii) if $P_{C_1} =$ # and $P_{C_2} \neq$ # then $P_C(d) = max(\{P_{C_2}(d'') : d' \in D', d'' \in D'', d' + d'' = d\})$,
(iv) if $P_{C_1} \neq$ # and $P_{C_2} \neq$ # then $P_C(d) = max\left(\left\{min\left(P_{C_1}(d'), P_{C_2}(d'')\right) : d' \in D', d'' \in D'', d' + d'' = d\right\}\right)$. ■

Ex.5. As an example, the composition between the constraint between RS and CCA and the constraint between CCA and $DGAb_s$ in Fig. 7 gives as result RS <(1, 0.0571429, 1), (2, 0.114286, 1), (3, 0.142857, 1), (4, 0.142857, 1), (5, 0.142857, 0.75), (6, 0.142857, 0.75), (7, 0.142857, 0.5), (8, 0.0857143, 0.5), (9, 0.0285714, 0.25)> $DGAb_s$. ■

Complexity (intersection and composition). Let D_1 and D_2 be two P+PQTLs, consisting of $|D_1|$ and $|D_2|$ distances respectively. Both intersection and composition take advantage of the fact that the distances between two time points are sorted. The intersection of D_1 and D_2 consists of at most $min(|D_1|, |D_2|)$ distances, and is evaluated in $T_{\cap_{pP}} = O\left(max(|D_1|, |D_2|)\right)$ time. The composition of D_1 and D_2 consists of at most $O(|D_1| + |D_2|)$ distances, and is evaluated in $T_{@_{pP}} = O(|D_1| * |D_2|)$ time.

5.2 Temporal reasoning with probabilities and preferences

In our approach, the goal of temporal reasoning is to compute the minimal network of a P+PQTN, propagating also preferences and probabilities. To achieve it, we apply a version of the Floyd-Warshall's algorithm (or one of its recent optimized versions, e.g., (Planken et al., 2008)), in which we replace $\cap$ and @ by our operators $\cap_{pP}$ and $@_{pP}$, respectively.

Complexity (temporal reasoning algorithm). Given the complexity of Floyd-Warshall's algorithm, and the above analysis of complexity of $\cap_{pP}$ and $@_{pP}$, the overall complexity of our temporal reasoning algorithm is $O\left(|V|^3 * max\left(T_{\cap_{pP}}, T_{@_{pP}}\right)\right)$, where $|V|$ is the number of time points in a P+PQTN, and $T_{\cap_{pP}}$ and $T_{@_{pP}}$ denote the time to evaluate $\cap_{pP}$ and $@_{pP}$, respectively, as defined above.

On the limit, $\max\left(T_{\cap_{pP}}, T_{@_{pP}}\right) = T_{@_{pP}}$. It is also easy to notice that $T_{@_{pP}}$ depends on the cardinalities (i.e., the number of distances) of the considered constraints ($|D_1|$ and $|D_2|$ above). To provide an accurate analysis of $T_{@_{pP}}$, we thus need to estimate the maximum cardinality of a constraint during the propagation process. Let us define $Dmax$ as the input constraint with the maximum cardinality before the propagation. In STP, and consequently in our algorithm, the application of composition operations between adjacent constraints in the network can produce constraints with increasing cardinality. Each composition can generate a constraint with at most a cardinality equal to the sum of the cardinalities of the composed constraints. Considering that the longest simple path in a network contains at most $|V| - 1$ constraints, the maximum cardinality of constraints during/after the propagation is thus $O(|Dmax| * |V|)$. As a consequence, $T_{@_{pP}} = O((|Dmax| * |V|)^2)$.

We can now formulate the overall complexity of our temporal reasoning algorithm as $O(|V|^3 * (|Dmax| * |V|)^2) = O(|V|^5 * |Dmax|^2)$.

In Fig.8 in the following, we show part of the minimal network obtained by our extended temporal reasoning algorithm applied to the constraints in Ex.1+Ex.1' (the complete minimal network is reported in Appendix A). Notably, while the "crisp" temporal constraints are the same as in Fig.5, in our approach such constraints are enriched by additional pieces of information concerning preferences and probabilities. For instance, while from the minimal network in Fig.5 we have that $INAGA_s$ can occur between 5 units before and 12 units after $DGAb_s$, we now provide physicians with additional relevant pieces of information about the preferences and probabilities. For instance, the most probable scenarios are the ones where the actual distance between $INAGA_s$ and $DGAb_s$ ranges between 1 and 5 time units (while the other ones are unlikely), and the scenarios associated with the higher

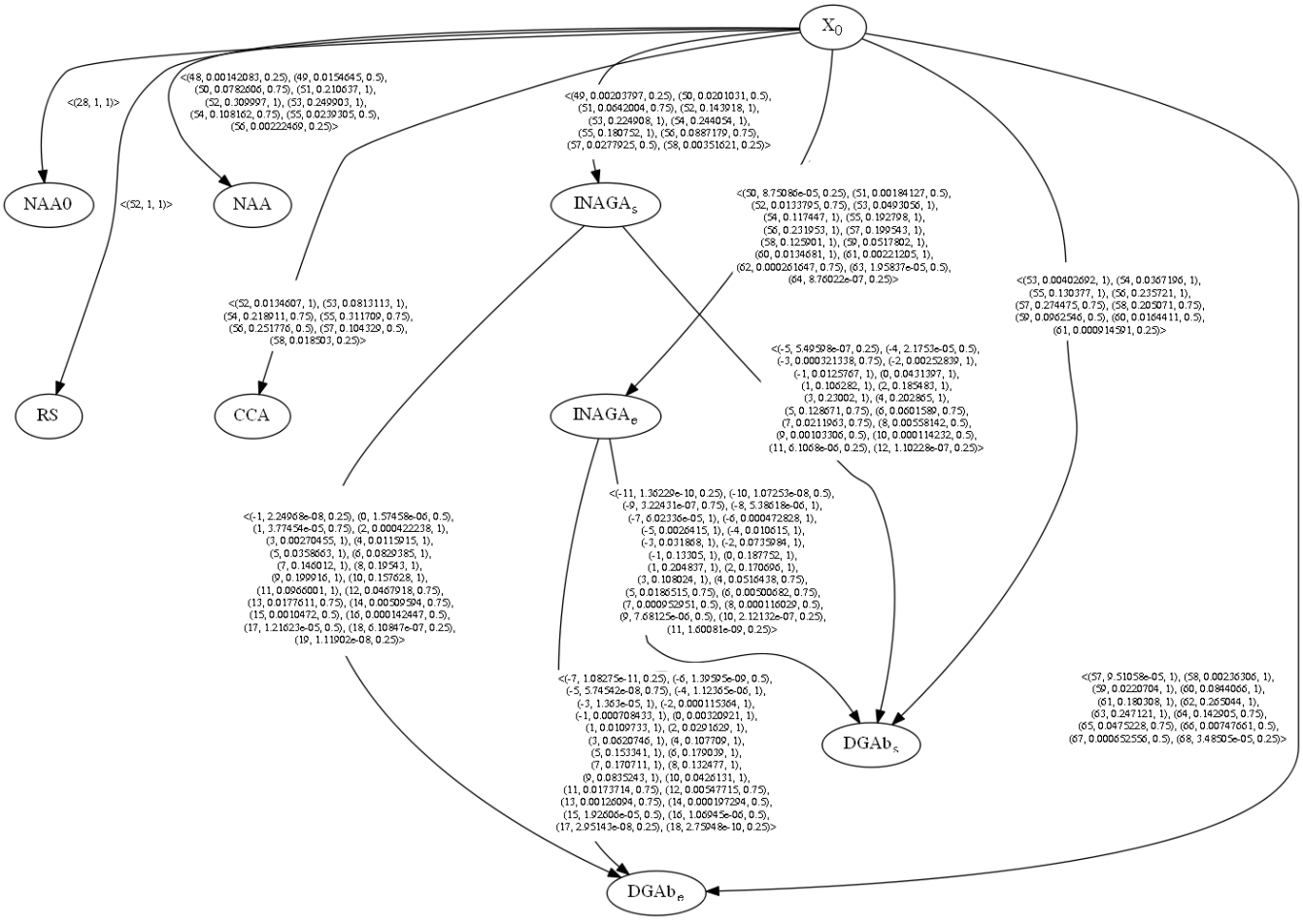

Fig. 8 Part of the P+PQTN minimal network for Ex.1, representing the temporal constraints between the actions/effects and the reference time X0, and the temporal constraints between the interacting effects.

preferences for the actions are the ones where such a distance ranges between -2 and 4 time units. However, the minimal network, and the preferences and probabilities it contains, are not easily consultable by physicians in a direct way. As a consequence, we provide a rich query language, and a graphical interface, to support physicians in the analysis of the results of temporal reasoning (see Section 6).

5.3 Experimental Evaluation

In this section, we present the results of the experimental evaluation we carried out to assess the performance of our temporal reasoning approach as well as its scalability with respect to the number of time points in a P+PQTN. To this end, we developed a prototype of our approach in C++ and we evaluated its performance on a Lenovo Thinkpad P1 Gen 3 equipped with a 2.6 GHz Intel Core i7-10750H CPU with 6 cores and 64 GiB of RAM, and running the Linux kernel version 5.13.12. To foster research and provide reproducibility of results we published the artifacts of our experimental evaluation on a public repository.[3]

[3] URL of the public repository of the artifacts of our experimental evaluation: https://gitlab.di.unipmn.it/sguazt/ppqtn-eswa2021-artifacts.

To evaluate the scalability of our temporal reasoning algorithm, we considered P+PQTNs of different sizes, ranging from 10 to 250 time points. Moreover, in the absence of a real dataset, for each P+PQTN we randomly generated its constraints as follows: distance values are drawn from a uniform probability distribution over the integer interval [0,100], probability and preference values are drawn from two different uniform distributions over the real interval [0,1], and we use two binomial distributions with parameter 0.5 to decide whether a constraint has defined probability and preference values or not, respectively. Finally, we used as performance metric the average execution time, i.e., the amount of time that our approach used the CPU to produce the minimal network for an input P+PQTN. This average value has been computed by running our approach several times over randomly generated P+PQTNs with the same number of time points and with randomly generated preferences and probabilities, and by averaging the execution time, until the relative precision of its 95% confidence interval became ≤ 4% (Banks et al., 2010).

We present the results of our experiments in Fig. 9 where each black-filled circle denotes the average execution time taken by our approach to process P+PQTNs with a given number of time points, and the error bars denote the associated 95% confidence interval. To study the order of growth of the execution time, we fit both polynomial and exponential regression models to the obtained results (where we used the number of time points as the predictor (inde-

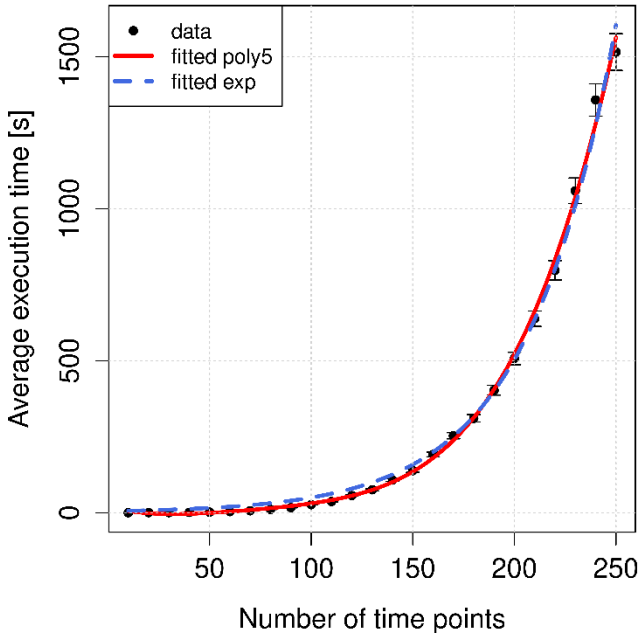

Fig. 9 Execution time (in seconds) of our approach as a function of the number of time points in a P+PQTN.

pendent) variable and the average execution time as the response (dependent) variable), and, for each one of them, we evaluated its goodness-of-fit by analyzing its residuals and by considering as goodness-of-fit measures the adjusted R^2, the Root Mean Square Error (RMSE) and the Akaike Information Criterion (AIC), which are typically used for model selection (e.g., see (Ding et al., 2018; Emmert-Streib & Dehmer, 2019)). From this analysis, we found that a 5th degree polynomial model provides a better fit than the exponential model. This finding can also be observed in Fig. 9 and in Table 1. Specifically, in Fig. 9 we note that the fitted 5th degree polynomial function (see the red solid line labeled as "fitted poly5") better fits the experimental data than the fitted exponential function (see the blue dashed line labeled as "fitted exp"), especially for a number of time points greater than 240. Also, from Table 1, we note that the fitted 5th degree polynomial model yields better values of the considered goodness-of-fit measures than the fitted exponential model; in particular, it results in lower RMSE and AIC values (intuitively, the lower, the better) as well as in a higher adjusted R^2 value (intuitively, the closer to one, the better).

Table 1. Goodness-of-fit measures of the 5th degree polynomial and the exponential regression models fitted to the experimental data.

Fitted model	**Goodness-of-fit measures**		
	Adjusted R^2	**RMSE**	**AIC**
Fitted poly5	0.997	24.65	238.32
Fitted exp	0.995	31.54	247.43

These results confirm the complexity analysis provided in the previous section. For example, the average execution

time required to deal with a P+PQTN of 100, 150 and 200 time points is 26.17 (standard deviation 5.19), 138.30 (standard deviation 27.81), and 507.35 (standard deviation 92.80) seconds, respectively.

5.4 Extending GLARE SSCPM to cope with Probabilities and Preferences

Following the approach in (Anselma et al., 2017), and taking advantage of the P+PQTNs approach described above, we have extended the Interaction Detection and the Temporal reasoner modules of GLARE-SSCPM to cope with probabilities and preferences. Such an extension has required several operations.

First, we have extended the knowledge models (both CIGs and ontological model) to represent probabilities and preferences. As regards the CIGs, we have extended GLARE's formalism to include preferences in the representation of temporal constraints between actions. As a simple example, the constraint between two consecutive administrations of Nalidixic Acid NAA0 and NAA was previously modeled in GLARE as NAA0<[20,28]>NAA. In the extended formalism, the same constraint is now extended to include the preferences explained in Ex.1+Ex.1': NAA0<(20,0.25), (21,0.50), (22,0.75), (23,1), (24,1), (25,1), (26,0.75), (27,0.50), (28,0.25)>NAA, where the first element of each pair is a possible delay (expressed in time units) and the second one is its preference. Similarly, we extended the ontological model of GLARE-SSCPM to represent the probabilities of the temporal constraints between actions and effects. In the previous ontological model of GLARE-SSCPM, each "hasEffect" relationship was including the representation of the temporal constraints between the action/drug and the related effect. As an example, the "hasEffect" relationship between "Nalidixic Acid" and the effect "Increase of Nalidixic Acid Gastric Absorption" shown in Fig. 3 contains the constraints NA<[1,2]>$INAGA_s$ and NA<[2,8]>$INAGA_e$. In the extended formalism, such constraints are extended to model probabilities (i.e., they are replaced by NA<(1,0.4), (2,0.6)>$INAGA_s$ and NA<(2,0.15), (3,0.25), (4,0.25), (5,0.15), (6,0.10),(7,0.05),(8,0.05)>$INAGA_e$, where the first element of each pair is a delay, and the second one is the probability of such a delay).

Since we only extended the representation of the constraints, and we have not added "ex-novo" any knowledge source, we still exploit the optimized constraint extraction procedure described in (Anselma et al., 2017): when an interaction has to be analyzed, the procedure retrieves from the ontological model, the CIGs and the log only the useful constraints. Since the extracted constraints are not in the form of P+PTQCs, only a minor extension has been needed to transform the extracted constraints into P+PTQCs constraints. Specifically:

- for constraints extracted from the ontological model, which are "purely" probabilistic, we add undefined preferences: the procedure assigns preference # to each possible temporal value within the constraint;

  For instance, the procedure translates the constraint NA<(1,0.4), (2,0.6)>$INAGA_s$, extracted to the ontological model, to NA<(1,0.4,#), (2,0.6,#)>$INAGA_s$.

- for constraints extracted from the CIGs, which are "purely" preferential, we add undefined probability: the procedure assigns probability % for each temporal value within the constraint;

  For instance, the procedure translates the constraint NAA0<(20,0.25), (21,0.50), (22,0.75), (23,1), (24,1), (25,1), (26,0.75), (27,0.50), (28,0.25),>NAA, extracted from the CIG, to NAA0<(20,%,0.25), (21,%,0.50), (22,%,0.75), (23,%,1), (24,%,1), (25,%,1), (26,%,0.75), (27,%,0.50), (28,%,0.25),>NAA.

- for constraints extracted from the log, the given preference is #. The probability is 1 if the log is precise (i.e., each constraint contains only one temporal value), or it is % if the log is imprecise (i.e., the time of execution of an action is not completely known).

  For instance, a precise log constraint X_0<(28)>NAA0 is translated to X_0<(28,1,#)>NAA0.

Then, we have extended the GLARE-SSCPM Interaction Detection module to adopt the temporal reasoning algorithm presented in Section 5.2 and to allow users to ask a set of queries (see Section 6) to analyze preferences and probabilities. Finally, we have extended the GLARE-SSCPM GUI to graphically show the analysis of the interactions (see Section 6).

6 Query Answering Facilities

Given a P+PQTN, our approach evaluates the *minimal network*, which is an implicit compact representation of the space of all possible solutions. However, minimal networks are usually quite large and complex (even in case they regard "crisp" constraints only), so that it may be difficult, for non-expert users, to read and interpret them (consider, e.g., the minimal network in Fig.8 and Appendix A, which corresponds to the example Ex.1+Ex.1' in Fig.7). Therefore, we think that providing facilities to explore\query the output of temporal reasoning is an important step, to practical-

ly support physicians. We thus provide users with query answering facilities. Notably, our goal in this context is to provide users with a "general" and flexible query language, supporting four different types of analysis (<ExtractionQ>, <FilterPPQ>, <DistQ>, and <QualitQ>). Additionally, we also support hypothetical queries (<HypothQ>), which are queries of any of the four types above, to be answered in a context in which a set of P+PQTCs is assumed.

In particular, <QualitQ> queries about the qualitative constraint INTERSECT (see below) are the ones that can be used (possibly in a hypothetical context) by physicians to look for the temporal intersections between the effects of guideline actions (i.e., to check for possible interactions, and their preferences\probabilities). Additionally, hypothetical queries can be exploited by physicians to perform "*what-if*" temporal analysis.

Part of the (extended) BNF grammar of our query language is shown below:

<Query> ::= <HypQ> | <StandardQ>

<HypQ> ::= <StandardQ> IF <(P+PQTC)$^+$>

<StandardQ> ::= <ExtractionQ> | <FilterPPQ>| <DistQ> | <QualitQ>

<ExtractionQ> ::= <point> ? <point> |<point> ? <point> ; <ExtractionQ>

<FilterPPQ> ::= prob <Op> <val> pref <Op> <val> ?

<Op> ::= < | = | > | ≥ | ≤

<DistQ> ::= <point> <PPCList> <point>

<PPCList> ::= <distance> prob <Op> <val> pref <Op> <val> |

<distance> prob <Op> <val> pref <Op> <val>, <PPCList>

<QualitQ> ::= <point> <PQRel> <point> ?| <point> <PIQRel> <interval> ?| <interval> <IQRel> <interval> ?

<PQRel> ::= < | = | > | ≠ | ≥ | ≤

<PIQRel> ::= •BEFORE | •MEETS |… | •AFTER

<IQRel> ::= BEFORE | MEETS | … | AFTER | DISJOINT | INTERSECT

Notably, <val> stands for a numeric real value in the interval [0,1], while <point> and <interval> stands for instantaneous (e.g., CCA) vs. durative (e.g. DGAb) actions\facts.

6.1 Basic Extraction Queries

Basic extraction queries (<ExtractionQ>) ask for the temporal distances between a set of pairs of time points, and for the probabilities and preferences of such distances. The output of such queries is simply obtained by retrieving the temporal constraints between the pairs of points from the minimal network.

Ex.6. $INAGA_e$? $DGAb_s$ asks the inferred constraints between the start of DGAb and the end of INAGA. The answer is $INAGA_e$ <(-11, 1.36229e-10, 0.25), (-10, 1.07253e-08, 0.5), (-9, 3.22431e-07, 0.75), (-8, 5.38618e-06, 1), (-7, 6.02336e-05, 1), (-6, 0.000472828, 1), (-5, 0.0026415, 1), (-4, 0.010615, 1), (-3, 0.031868, 1), (-2, 0.0735984, 1), (-1, 0.13305, 1), (0, 0.187752, 1), (1, 0.204837, 1), (2, 0.170696, 1), (3, 0.108024, 1), (4, 0.0516438, 0.75), (5, 0.0186515, 0.75), (6, 0.00500682, 0.75), (7, 0.000952951, 0.5), (8, 0.000116029, 0.5), (9, 7.68125e-06, 0.5), (10, 2.12132e-07, 0.25), (11, 1.60081e-09, 0.25)> $DGAb_s$.

Ad-hoc facilities (not shown in the BNF grammar) are provided to facilitate the analysis of the relative temporal position of ***durative*** actions\facts. For instance, the query in Ex.6BIS is a shorthand for the query "X_0 ? $DGAb_s$, X_0 ? $DGAb_e$, X_0 ? $INAGA_s$ and X_0 ? $INAGA_e$", and asks for the temporal location of the endpoints of the durative facts DGAb and INAGA with respect to the reference time X_0 (of course, such an extraction query is useful to analyse the possible temporal intersection between DGAb and INAGA).

Ex.6BIS the answer to the query DGAb ?? INAGA, considering the constraints described in Ex.1+Ex.1', is {X_0 <(53, 0.00402692, 1), (54, 0.0367196, 1), (55, 0.130377, 1), (56, 0.235721, 1), (57, 0.274475, 0.75), (58, 0.205071, 0.75), (59, 0.0962546, 0.5), (60, 0.0164411, 0.5), (61, 0.000914591, 0.25)> $DGAb_s$, X_0 <(57, 9.51058e-05, 1), (58, 0.00236306, 1), (59, 0.0220704, 1), (60, 0.0844066, 1), (61, 0.180308, 1), (62, 0.265044, 1), (63, 0.247121, 1), (64, 0.142905, 0.75), (65, 0.0475228, 0.75), (66, 0.00747661, 0.5), (67, 0.000652556, 0.5), (68, 3.48505e-05, 0.25)> $DGAb_e$, X_0 <(49, 0.00203797, 0.25), (50,

0.0201031, 0.5), (51, 0.0642004, 0.75), (52, 0.143918, 1), (53, 0.224908, 1), (54, 0.244054, 1), (55, 0.180752, 1), (56, 0.0887179, 0.75), (57, 0.0277925, 0.5), (58, 0.00351621, 0.25)> $INAGA_s$, X_0 <(50, 8.75086e-05, 0.25), (51, 0.00184127, 0.5), (52, 0.0133795, 0.75), (53, 0.0493056, 1), (54, 0.117447, 1), (55, 0.192798, 1), (56, 0.231953, 1), (57, 0.199543, 1), (58, 0.125901, 1), (59, 0.0517802, 1), (60, 0.0134681, 1), (61, 0.00221205, 1), (62, 0.000261647, 0.75), (63, 1.95837e-05, 0.5), (64, 8.76022e-07, 0.25)> $INAGA_e$}.

We have extended the graphical interface of GLARE-SSCPM to be able to graphically show the output of extraction queries (including preferences and probabilities). Fig. 10 show the enhanced representation, where each distance is paired with its probability[4] (upper number) and its preference (lower number). Comparing Fig.10 with Fig. 6 above, it is easy to notice the additional informative content. Indeed, as regards probabilities, one can notice that DGAb has a high probability to start around time 57, and to end around time 62. On the other hand, INAGA has a high probability to start around time 53-54, and to end around time 56-57. On the other hand, as concern preferences, we have that, e.g., scenarios in which DGA starts in [53,56] and ends in [57,63] have preference 1.

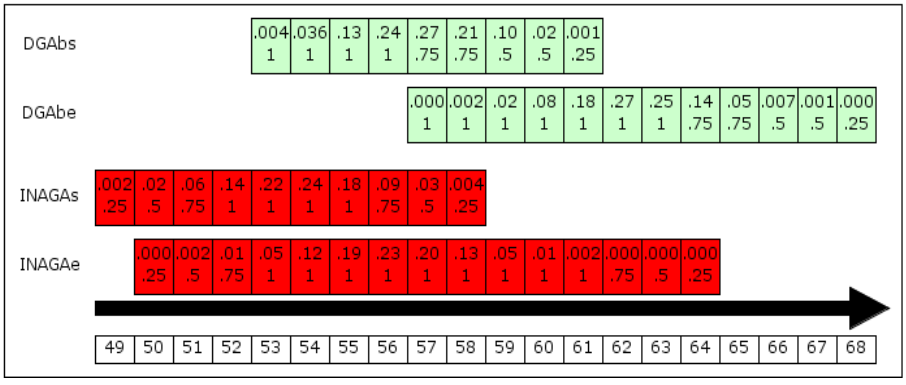

Fig. 10 GUI representation of the output of the query DGA ?? INAGA in our extended approach. Each line represents the time interval when the starting or ending point of an effect may occur. For each time point, probability (upper number) and preference (lower number) are shown.

Notably, though extraction queries (and the graphical representation of their output) facilitate users in the analysis of the output of temporal reasoning (i.e., of the minimal network) in general, and of possible temporal interactions (i.e., intersections of effects) in particular, still it is not easy, for physicians, to analyse\infer the preferences and probabilities of the different scenarios. For such a purpose, we support other types of queries, and, in particular, qualitative queries (see subsection 6.4).

6.2 Filter Queries

Filter probability+preference queries (<FilterPPQ>) can be used to focus on distances whose preference and/or probability satisfy a given condition (e.g. distances with probability >0.05 and preference >0.2). They provide as output the P+PQTN obtained by removing from the constraints all those triples *(d,p,P)* that do not satisfy the conditions.

6.3 Boolean queries about distances

Boolean queries about distances (<DistQ>) ask whether the input distances between two points are possible given the minimal network, and their preferences and probabilities satisfy the conditions in the query, and return a boolean value. To answer a <DistQ> query $t_i\langle d_1 \text{prob} 0\text{p}_1 p_1 \text{pref} 0\text{p}_2 P_2\rangle t_j$ (e.g., "$t_i\langle 3\text{prob} > 0.1 \text{pref} > 0.2\rangle t_j$") our approach retrieves the constraint between t_i and t_j from the minimal network and checks whether the distance between t_i and t_j may be d_1, and whether its probability and preference satisfy the conditions.

6.4 Qualitative Queries

Qualitative Queries (<QualitQ>). Though our approach grounds on STP quantitative (metric) temporal constraints, at the query level we can also support queries asking for the probability and the preference of qualitative constraints between instantaneous events (i.e., points) and\or durative ones. In particular, we consider:

(i) the point—point relationships of Vilain and Kautz's algebra (M. B. Vilain & Kautz, 1986) –see <PQRel> in the

[4] Probabilities of .000 represents time points with a probability less than 0.001 that, due to representation constraints, cannot be correctly visualized.

grammar

(ii) the point—interval basic relationships of Vilain's Algebra (M. Vilain, 1982) – see <PIQRel> in the grammar, and

(iii) the interval—interval basic relationships of Allen's interval Algebra (Allen, 1983), – see <IQRel> in the grammar.

Since in the minimal network we have only distances between points (and their preferences and probabilities), we have to:

(1) take into account the distances that "satisfy" the input qualitative conditions, and

(2) derive the output preferences and probabilities of the conditions on the basis of the preferences and probabilities of such distances.

In the following, we propose the rules to achieve such a goal, considering probabilities, and then preferences.

Definition. ***Probabilities of relationships between time points***. Given two time points t_1 and t_2, and given the temporal constraint t_1 $<(d_1,p_1,P_1), \ldots,(d_k,p_k,P_k)>$ t_2, we indicate with $\varphi(d_i)$ the probability of the distance d_i (i.e., $\varphi(d_i)=p_i$); thus, we define the probability $Prob(t_1\ op\ t_2)$ of relationship *op* between time points t_1 and t_2 (i.e., of the relationships <PQRel> in the grammar) as follows:

$Prob(t_2 > t_1) = \sum_{d_i>0} \varphi(d_i)$ if $\exists d_i \in \{d_1,\ldots,d_k\}$ such that $d_i>0$, 0 otherwise

$Prob(t_2 = t_1) = \varphi(0)$ if $0 \in \{d_1,\ldots,d_k\}$, 0 otherwise

$Prob(t_2 < t_1) \sum_{d_i<0} \varphi(d_i)$ if $\exists d_i \in \{d_1,\ldots,d_k\}$ such that $d_i<0$, 0 otherwise

$Prob(t_2 \geq t_1) = Prob(t_2 > t_1) + Prob(t_2 = t_1)$

$Prob(t_2 \neq t_1) = Prob(t_2 > t_1) + Prob(t_2 < t_1)$

$Prob(t_2 \leq t_1) = Prob(t_2 < t_1) + Prob(t_2 = t_1)$ ■

Definition. ***Preferences of relationships between time points***. Given two any time points t_1 and t_2, and given the temporal constraint t_1 $<(d_1,p_1,P_1), \ldots,(d_k,p_k,P_k)>$ t_2, we indicate with $\sigma(d_i)$ the preference of the distance d_i (i.e., $\sigma(d_i)=P_i$); thus, we define the preference $Pref(t_1\ op\ t_2)$ of relationship *op* between time points t_1 and t_2 (i.e., of the relationships <PQRel> in the grammar) as follows:

Let $D=\{d \mid d \in \{d_1,\ldots,d_k\} \wedge d>0\}$

$Pref(t_2 > t_1) = MAX_{d \in D}(\sigma(d))$ if $D \neq \emptyset$, 0 otherwise

Let $D=\{d \mid d \in \{d_1,\ldots,d_k\} \wedge d<0\}$

$Pref(t_2 < t_1) = MAX_{d \in D}(\sigma(d))$ if $D \neq \emptyset$, 0 otherwise

$Pref(t_2 = t_1) = \sigma(0)$ if $0 \in \{d_1,\ldots,d_k\}$, 0 otherwise

$Pref(t_2 \geq t_1) = MAX(Pref(t_2 > t_1), Pref(t_2 = t_1))$

$Pref(t_2 \neq t_1) = MAX(Pref(t_2 > t_1), Pref(t_2 < t_1))$

$Pref(t_2 \leq t_1) = MAX(Pref(t_2 < t_1), Pref(t_2 = t_1))$ ■

In our approach, durative events are coded as pairs of points, denoting the starting and ending time of their interval of occurrence (see Section 4). Allen's interval relationships are the most common tool used in AI to encode qualitative relationships between durative events. Each relationship can be easily mapped onto the corresponding conjunction of relationships between interval endpoints. We exploit such a mapping to define the probability and preference of Allen's relationships in terms of the probability and preference of the qualitative relationships between endpoints which encode them.

Besides Allen's relationships, we also consider the *INTERSECT* and the *DISJOINT* relationships, which are very useful in many domains, and, specifically, in the comorbidity context.

Definition. ***Probabilities of relationships between events***. Given two any (durative) events e_1 and e_2, and indicating by $start(e_i)$ and $end(e_i)$ their endpoints (e.g., if e_i is DGAb, then $start(e_i)$ is $DGAb_s$ and $end(e_i)$ is $DGAb_e$), we define:

$Prob(BEFORE(e_1,e_2)) = Prob((end(e_1)+1) < start(e_2))$

$Prob(MEETS(e_1,e_2)) = Prob((end(e_1)+1) = start(e_2))$

$Prob(OVERLAPS(e_1,e_2)) = Prob(start(e_1) < start(e_2)) * Prob(end(e_1) \geq start(e_2)) * Prob(end(e_1) < end(e_2))$

$Prob(ENDED\text{-}BY(e_1,e_2)) = Prob(start(e_1) < start(e_2)) * Prob(end(e_1) = end(e_2))$

$Prob(CONTAINS(e_1,e_2)) = Prob(start(e_1) < start(e_2)) * Prob(end(e_1) > end(e_2))$

$Prob(STARTS(e_1,e_2)) = Prob(start(e_1) = start(e_2)) * Prob(end(e_1) < end(e_2))$

$Prob(EQUAL(e_1,e_2)) = Prob(start(e_1) = start(e_2)) * Prob(end(e_1) = end(e_2))$

$Prob(STARTED\text{-}BY(e_1,e_2)) = Prob(start(e_1) = start(e_2)) * Prob(end(e_1) > end(e_2))$

$Prob(DURING(e_1,e_2)) = Prob(start(e_1) > start(e_2)) * Prob(end(e_1) < end(e_2))$

$Prob(ENDS(e_1,e_2)) = Prob(start(e_1) > start(e_2)) * Prob(end(e_1) = end(e_2))$

$Prob(OVERLAPPED\text{-}BY(e_1,e_2)) = Prob(start(e_1) > start(e_2)) * Prob(start(e_1) \leq end(e_2)) * Prob(end(e_1) > end(e_2))$

$Prob(MET\text{-}BY(e_1,e_2)) = Prob((end(e_2)+1) = start(e_1))$

$Prob(AFTER(e_1,e_2)) = Prob((end(e_2)+1) < start(e_1))$

$Prob(DISJOINT(e_1,e_2)) = Prob(BEFORE(e_1,e_2)) + Prob(MEETS(e_1,e_2)) + Prob(MET\text{-}BY(e_1,e_2)) + Prob(AFTER(e_1,e_2))$

$Prob(INTERSECT(e_1,e_2)) = 1 - Prob(DISJOINT(e_1,e_2))$ ■

Definition. ***Preferences of relationships between events***. Given two any (durative) events e_1 and e_2, and indicating by $start(e_i)$ and $end(e_i)$ their endpoints, we define:

$Pref(BEFORE(e_1,e_2)) = Pref((end(e_1)+1) < start(e_2))$

$Pref(MEETS(e_1,e_2)) = Pref((end(e_1)+1) = start(e_2))$

$Pref(OVERLAPS(e_1,e_2)) = MIN(Pref(start(e_1) < start(e_2)), Pref(end(e_1) \geq start(e_2)), Pref(end(e_1) < end(e_2)))$

$Pref(ENDED\text{-}BY(e_1,e_2)) = MIN(Pref(start(e_1) < start(e_2)), Pref(end(e_1) = end(e_2)))$

$Pref(CONTAINS(e_1,e_2)) = MIN(Pref(start(e_1) < start(e_2)), Pref(end(e_1) > end(e_2)))$

$Pref(STARTS(e_1,e_2)) = MIN(Pref(start(e_1) = start(e_2)), Pref(end(e_1) < end(e_2)))$

$Pref(EQUAL(e_1,e_2)) = MIN(Pref(start(e_1) = start(e_2)), Pref(end(e_1) = end(e_2)))$

$Pref(STARTED\text{-}BY(e_1,e_2)) = MIN(Pref(start(e_1) = start(e_2)), Pref(end(e_1) > end(e_2)))$

$Pref(DURING(e_1,e_2)) = MIN(Pref(start(e_1) > start(e_2)), Pref(end(e_1) < end(e_2)))$

$Pref(ENDS(e_1,e_2)) = MIN(Pref(start(e_1) > start(e_2)), Pref(end(e_1) = end(e_2)))$

$Pref(OVERLAPPED\text{-}BY(e_1,e_2)) = MIN(Pref(start(e_1) > start(e_2)), Pref(start(e_1) \leq end(e_2)), Pref(end(e_1) > end(e_2)))$

$Pref(MET\text{-}BY(e_1,e_2)) = Pref((end(e_2)+1) = start(e_1))$

Pref(AFTER(e_1,e_2)) = Pref((end(e_2)+1) < start(e_1))

Pref(DISJOINT(e_1,e_2)) = MAX(Pref(BEFORE(e_1,e_2)), Pref(MEETS(e_1,e_2)), Pref(MET-BY(e_1,e_2)), Pref(AFTER(e_1,e_2)))

Pref(INTERSECT(e_1,e_2)) = MAX(

Pref(OVERLAPS(e_1,e_2)), Pref(ENDED-BY(e_1,e_2)), Pref(CONTAINS(e_1,e_2)), Pref(STARTS(e_1,e_2)), Pref(EQUAL(e_1,e_2)), Pref(STARTED-BY(e_1,e_2)), Pref(DURING(e_1,e_2)), Pref(ENDS(e_1,e_2)), Pref(OVERLAPPED-BY(e_1,e_2))) ■

For the sake of brevity, we omit the definition of Vilain's qualitative relationships between a time point and a time interval (M. Vilain, 1982), which is very similar.

On the basis of the above definitions, the preferences and probabilities of qualitative relationships can be trivially derived from the constraints in minimal network, to answer queries.

Note.2. The interpretation of the *probability* of qualitative relations is quite natural and intuitive. For example, the probability of "INAGA INTERSECT DGAb" (see Ex.8 below) is the probability of the temporal scenarios in which "INAGA INTERSECT DGAb" holds (computed on the basis of the probabilities of the (propagated) distances such that "INAGA INTERSECT DGAb" holds; see definition of *Prob(INTERSECT(e_1,e_2))* above). On the other hand, we have to explain the correct interpretation of *preferences* in the comorbidity context. Indeed, since the interaction between INAGA and DGAb is not desirable, and only happens in case INAGA and DGAb temporally intersect, it might look odd to speak about the "preference" on the relationship "INAGA INTERSECT DGAb" (or, alternatively, one may argue that its preference is null).

Indeed, it is important to point out that:

(i) as mentioned in Note.1 in the introduction, *input* preferences concern the possible times when physicians can perform CIG actions, and are evaluated in each CIG, independently of the others (for instance, considering Ex.1+Ex.1', the preferences of the possible execution times of CCA -- in our approach, execution times are expressed as distances between CCA and the reference time -- are evaluated considering on the basis of pharmacokinetics and pharmacodynamics studies, and without taking into consideration general issues such as the fact that, for specific patients, CCA may be combined with NAA), and

(ii) the preferences we deal with in our approach only *concerns temporal distances*, independently of how much the events occurring at such distances are desirable or not.

Therefore, the preference for the INTERSECT relation between INAGA and DGAb has not to be interpreted as the preference for the fact that there is an intersection, and thus a possible interaction, but as the preference of the "temporal scenario" in which there is an intersection. Roughly speaking, such a preference is computed as discussed in the definition of *Pref(INTERSECT(e_1,e_2))* above, on the basis of the preferences of the distances between $INAGA_s$, $INAGA_e$, $DGAb_s$, and $DGAb_e$, which, in turn, result from the propagation of the preferences of the distances of NAA and CCA from X_0, combined with the other input distances (see Fig.2). *Thus, in the example in Fig.7, the preference of "INAGA INTERSECT DGAb" is the **temporal** cumulative preference of the scenarios in which "INAGA INTERSECT DGAb" holds, evaluated in terms of the combined preferences of the times when the actions NAA (which leads to INAGA) and CCA (which leads to DGAb) can be executed*. ■

Ex.7. For example, the query "$INAGA_e$ < $DGAb_s$?" asks for the preference and the probability of the case in which INAGA ends before the start of DGAb. The answer is a probability of 0.559936 and a preference of 1. ■

Ex.8. The query "INAGA INTERSECT DGAb ?" asks for the preference and the probability of the case in which INAGA intersects DGAb. Notably, such a query can be used by physician to derive, from the minimal network, the probability of the intersection (notably, this is the case in which the two effects may interact), and the degree of the preference of such situation. The answer to this query is a probability of 0.440064 and a preference of 1. ■

6.5 Hypothetical queries

Hypothetical queries (<HypQ>) are composite queries, in which a set of P+PQTC constraints has to be assumed (hypothesized), before asking a standard query <StandardQ>.

Such queries are answered in several steps. Indeed, temporal reasoning is needed in order to see whether the constraints in the hypothesis are consistent with the ones in the minimal network, and to propagate their implications.

(1) First, we (provisionally) add the new P+PQTC constraints to the minimal network (using the operation of inter-

section ($\cap_{pP}$)).

(2) We then propagate the new set of constraints through our version of Floyd-Warshall's algorithm. In case the propagation detects an inconsistency, the output is a warning, stating that the constraints in the hypothesis are not consistent with the minimal network (i.e., they are not possible, given the input constraints). Otherwise, the new minimal network MN_{Hyp} obtained by the propagation is considered.

(3) Finally, <StandardQ> is answered (as detailed above) in MN_{Hyp}.

Ex. 9. "$INAGA_e$? $DGAb_s$ IF {X_0 <(53,%,#),(54,%,#)> CCA, X_0 <(48,%,#),(49,%,#)> NAA}" asks the constraints between the start of DGAb and the end of INAGA in case CCA is executed in units 53 or 54, and NAA in units 48 or 49. The answer is $INAGA_e$ <(-3, 1.74465e-58, 0.5), (-2, 5.94715e-30, 0.5), (-1, 2.02678e-11, 0.5), (0, 0.0934398, 0.5), (1, 0.906557, 0.5), (2, 3.43289e-06, 0.5), (3, 5.88096e-21, 0.5), (4, 3.85778e-45, 0.5), (5, 1.0065e-84, 0.5), (6, 3.2115e-132, 0.5), (7, 4.42862e-200, 0.25)> $DGAb_s$. ■

Notably, also the output of extraction hypothetical queries can be shown by the extended graphical interface.

6.6 An interactive session of temporal analysis

Our approach provides physicians with a complete and flexible tool for the temporal analysis of the interactions between CIGs. In this section, we describe a session of analysis of the interaction between NAA and CCA described in Ex.1 and Ex.1' using some of the queries described above. In our example, we suppose that first (STEP1) the physician decides to analyse the interaction "as it is", using only the information contained in the log, the CIGs and the ontological model. Then (STEP2), she decides to make some hypotheses on the times of execution of the two actions. In particular, she analyses the interaction hypothesizing that NAA and CCA are executed at times with maximum preference. Since the given result is not satisfying in terms of probability of the interaction (i.e., it is too high), in STEP3 and STEP4, we suppose that physician hypothesizes to move in time CCA and NAA to find a configuration with a low probability of interaction, but still with a good preference value. It is worth stressing that the steps described below are only illustrative: the physician is free to follow her-own criteria and reasoning process to investigate the possible temporal interactions between CIG actions.

STEP1: at the beginning of the session, the physician wants to analyse the interaction considering only the information contained in the log, the CIGs and the ontological model. First, she asks for the graphical representation of the query "DGAb ?? INAGA", and our tool shows to physician the graphical representation shown in Fig.10 above. As already explained above, the GUI allows the physician to easily notice that the interaction is possible, and to evaluate the probabilities and the preferences associated to each time point. To better understand the probability of the interaction, the physician asks for the query "DGAb INTERSECT INAGA ?", having as result a probability of 0.440064 with preference 1.

STEP2: then, we suppose that the physician wants to analyse the same interaction, but in the hypothesis that CCA and NAA are executed at times with maximum preferences. In particular, she asks for the probability and preference in which INAGA intersects DGAb assuming that CCA is executed in units 52 or 53, and NAA in units 51 or 52 (query "INAGA INTERSECT DGAb ? IF {X_0 <(52,%,#), (53,%,#)> CCA, X_0 <(51,%,#), (52,%,#)> NAA}"). The answer to this query is that the intersection has probability 0.681259 and preference 1 (for the sake of brevity, we do not show the GUI for the analysis of such a hypothetical analysis). Intuitively, the probability of intersection is quite high, since CCA and NAA are executed closely, and, according to temporal distances and probabilities in the constraints in Fig. 7, this will result in a very likely interaction between INAGA and DGAb. The preference of the intersection scenario is, obviously, high, since it refers to high preferences for the times of execution of the actions. However, since the primary goal is to decrease the probability of the interaction (while possibly achieving a high preference), the given result is not satisfying.

STEP3: to decrease the probability of the interaction, the physician decides to postpone the execution of NAA. The choice falls on times (54 and 55), which have good preferences (0.75 and 0.5 respectively), to be as conformant as possible with the CIGs, but are "far enough" from the times chosen in STEP2. The asked hypothetical query is "INAGA INTERSECT DGAb ? IF {X0 <(52,%,#), (53,%,#)> CCA, X0 <(54,%,#), (55,%,#)> NAA}". The answer to this query is that the intersection has probability 0.999999 and preference 0.75. As can be easily noticed, the given configuration does not lead to better results. Indeed, the probability of the interaction is almost certain.

STEP4: given the result obtained in STEP3, the physician decides to backtrack, for NAA, to the times of execution of STEP2 (51 or 52), and instead to postpone the execution of CCA to times 56 or 57 (i.e., times with a preference of 0.5). The corresponding query is "INAGA INTERSECT DGAb ? IF {X_0 <(56,%,#),(57,%,#)> CCA, X_0 <(51,%,#),(52,%,#)> NAA}". The answer to this query is that the intersection has probability 5.0e-5 and preference 0.5. The preference of

this scenario is lower with respect to the ones in STEP2 and STEP3 (i.e., the scenario is less conformant to the CIG). However, the probability of the interaction is negligible, and the physician decides that this configuration is a good trade-off between the adherence to the CIG recommendations and the safety of the treatment for the specific patient.

The above example shows the effectiveness of our approach in integrating probability and preferences in temporal reasoning with respect to "crisp" approaches. As a matter of facts, our approach not only supports representation and reasoning about temporal preferences and probabilities, but also provides users with user-friendly facilities (queries and graphical visualizations) to explore such enriched temporal constraints, e.g., adopting a "what-if" methodology, and deeply taking advantage of the pieces of information provided by preferences and probabilities.

Notably, graphical support and hypothetical queries were already provided in the previous version of GLARE-SSCPM (see Section 3). However, in GLARE-SSCPM users would only be able to ask and visualize the "crisp" temporal constraints between the endpoints of INAGA and DGAb (possibly in a context in which a set of temporal constraints has been assumed), or Boolean queries asking whether INAGA and DGAb may intersect (possibly given a set of temporal constraints to be assumed). However, it is worth noticing that, in all the steps 1-4, the answer of the "crisp" version of GLARE-SSCPM (see Section 3) is simply that an intersection between INAGA and DGAb is *possible* (please, consider again footnote 1 in Section 3). This is, of course, a poorly informative answer, in case temporal preferences and probabilities are available. Instead, our new approach is more effective because, thanks to its ability to take into account also temporal preferences and probabilities, it allows physicians to perform more realistic and useful what-if analysis, to check not only if temporal interactions between actions are possible but also their probabilities and preferences.

7 CONCLUSIONS

CIG systems are widely adopted for medical knowledge-based decision support. Several recent approaches also extend the support to consider comorbid patients. For such patients, the detection of the possible interactions between the effects of the actions recommended by multiple CIGs (one for each disease of the patient) is of primary importance. Since, from the practical point of view, interactions can only occur between effects that *intersect* in time, a concrete analysis of interactions involves (i) the representation of the temporal constraints between CIG actions, and between such actions and their effects, and (ii) the propagation of such constraints. Until now, such issues have been faced, within the AIM community, only considering "crisp" temporal constraints (with the only exception of (Andolina et al., 2018), in which probabilistic temporal constraints have been taken into account). However, "crisp" temporal approaches are quite "rigid", since constraints may be either satisfiable or not. In the CIG context, several guidelines provide not only temporal constraints about the delay between actions, but also indicate that different delays have different *preferences*. Additionally, medical knowledge can be used to derive knowledge about the *probabilistic* distribution of the effects of CIGs' actions in time. Such preferences and probabilities, when available, can be exploited to provide physicians with a more flexible temporal mechanism (e.g., indicating not just whether a temporal intersection is necessary, possible or not possible – as in (Anselma et al., 2017) –, but also – in case it is possible – its probability, and the value of preference of the constraints in the scenario). To achieve such a result, our approach provides the following main contributions

(i) we have extended quantitative STP temporal constraints with both preferences and probabilities,

(ii) we have proposed an approach for propagating such temporal constraints (and their preferences and probabilities),

(iii) we have experimentally evaluated it,

(iv) we have proposed query-answering supports, and

(v) we have integrated our approach within an existing decision support system for the treatment of comorbidities (specifically, in GLARE-SSCPM).

Notably, our approach provides two main novelties with respect to the state of the art:

(i) we propose the *first* temporal framework in the literature able to represent, reason (propagate to determine the minimal network) and query "non-crisp" temporal constraints considering *both preferences and probabilities*.

(ii) We first adopt preferential+probabilistic temporal reasoning in the context of supporting physicians in the treatment of comorbid patients (through an integration of our temporal framework with the GLARE-

SSCPM approach)

We have also shown, with a running example, the application of our approach to a concrete case of comorbidity. In general, our temporal reasoning and query answering framework can be used to support run-time execution of CIGs on comorbid patients, to check for temporally possible interactions. At each time during the execution, physicians can trigger our temporal framework to check whether interactions may arise among the next actions to be executed in the guidelines. Probabilistic + preferential temporal reasoning is used to check not only whether interactions are temporally possible, but also their probabilities and preferences. Specifically, the check for possible intersections between effects can be useful at least in two different situations, during the run-time execution of multiple CIGs:

(1) *Focusing on the next actions in the different CIGs, physicians can check whether interactions are temporally possible, and their probability and preference.*

(2) *While choosing among alternative paths of actions in the CIGs, physicians can check which alternatives contain actions that may temporally interact with the other therapies currently in execution for the patient, considering also probabilities and preferences.*

In both cases, temporal reasoning can be used to propagate the temporal constraints and infer their minimal network, and query answering facilities can be used in order to directly check the probability and preferences of interactions. In particular, the hypothetical qualitative queries are very important in this context, since they support *what-if* analysis about the consequences of performing actions at given times (consider the case study described in Section 6.6).

Our current approach can be extended along several lines, and this will be the aim of our future work in the area. First, we plan to investigate the possibility to consider also dense and/or possibly non-contiguous values for distances. Second, several algorithms have been devised in order to optimize Floyd-Warshall's algorithm in the evaluation of STP minimal networks (consider, for instance, delta-STP (Lin Xu & Choueiry, 2003) or P3C (Planken et al., 2008)). In our future work, we will investigate the possibility of exploiting one of such optimizations also in our approach. Third, our main goal is that of providing users with decision support facilities, so that in the paper we address the need of providing them with a minimal network ("summarizing" all the possible solutions) and feasible facilities to explore it (e.g., through an "hypothesize and test" process). Nevertheless, an additional facility, providing users directly with a specific solution (the "optimal" one) could provide additional support. Several papers in the literature have already addressed the problem of finding an optimal solution for constraints with preferences (consider, e.g., (Khatib et al., 2001) as regards STP constraints with preferences). We would like to investigate whether the reasoning techniques proposed in such approaches can be adapted\extended in order to cope with our constraints (given a suitable definition of "optimality" considering both preferences and probabilities). Fourth, we aim at applying our approach also to different contexts and tasks. Indeed, in many AI tasks (e.g., planning, knowledge representation) and application areas, both "*endogenous*" and "*exogenous*" actions/events have to be managed together. Endogenous actions/events are actions/events to be executed by an agent; as a consequence, preferences on the execution times might be identified. On the other hand, exogenous events are events that occur "in the outside world", and agents have no control on them. Therefore, for such events, preferences on the execution times are meaningless. However, there might be probabilistic knowledge on the time of occurrence of such events. Though the primary goal of our approach has been to define an enriched temporal knowledge server for GLARE- SSCPM, to support physicians with a more flexible treatment comorbidities, the temporal approach in this paper is largely task and domain independent (indeed, only the query language has been designed, especially as regards qualitative queries, with a specific bias towards our medical application), and we plan to take advantage of it for other domains\tasks. Finally, as briefly discussed at the end of Section 2.1, a mainstream of research in temporal reasoning is aiming at extending the STN framework to cope also with other forms of uncertainty, leading to the STNU and CSTN families of frameworks. In a very recent work (Gao et al., 2020), Gao et al. have extended STNU to consider probabilities on uncertain durations, focusing on dynamic controllability. As future work, it might be worth to investigate whether some features of our approach (e.g., the representation of -temporal constraints with- preferences and probabilities, and the intersection and composition operations to propagate them) can be exploited\adapted to support the treatment of preferences and probabilities also in the STNU and CSTN frameworks, thus enhancing their generality and flexibility.

ACKNOWLEDGMENTS

This research has been partially supported by the Università del Piemonte Orientale.

REFERENCES

Adlassnig, K.-P., Combi, C., Das, A. K., Keravnou, E. T., & Pozzi, G. (2006). Temporal representation and reasoning in

medicine: Research directions and challenges. *Artif. Intell. Medicine*, *38*(2), 101–113. https://doi.org/10.1016/j.artmed.2006.10.001

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, *26*(11), 832–843. https://doi.org/10.1145/182.358434

Andolina, A., Anselma, L., Piovesan, L., & Terenziani, P. (2018). Querying Probabilistic Temporal Constraints for Guideline Interaction Analysis: GLARE's Approach. In G. R. Simari, E. Fermé, F. G. Segura, & J. A. R. Melquiades (Eds.), *Advances in Artificial Intelligence – IBERAMIA 2018 – 16th Ibero-American Conference on AI, Trujillo, Peru, November 13-16, 2018, Proceedings* (Vol. 11238, pp. 3–15). Springer. https://doi.org/10.1007/978-3-030-03928-8_1

Anselma, L., Piovesan, L., & Terenziani, P. (2017). Temporal detection and analysis of guideline interactions. *Artificial Intelligence in Medicine*, *76*, 40–62. https://doi.org/10.1016/j.artmed.2017.01.001

Anselma, L., Terenziani, P., Montani, S., & Bottrighi, A. (2006). Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines. *Artificial Intelligence in Medicine*, *38*(2), 171–195. https://doi.org/10.1016/j.artmed.2006.03.007

Augusto, J. C. (2005). Temporal reasoning for decision support in medicine. *Artif. Intell. Medicine*, *33*(1), 1–24. https://doi.org/10.1016/j.artmed.2004.07.006

Badaloni, S., & Giacomin, M. (2006). The algebra IAfuz: A framework for qualitative fuzzy temporal reasoning. *Artificial Intelligence*, *170*(10), 872–908. https://doi.org/10.1016/j.artint.2006.04.001

Banks, J., II, J. S. C., Nelson, B. L., & Nicol, D. M. (2010). *Discrete-Event System Simulation, 5th New Internatinal Edition*. Pearson Education.

Barták, R., Morris, R. A., & Venable, K. B. (2014). An Introduction to Constraint-Based Temporal Reasoning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *8*(1), 1–121. https://doi.org/10.2200/S00557ED1V01Y201312AIM026

Bilici, E., Despotou, G., & Arvanitis, T. N. (2019). Concurrent Execution of Multiple Computer-interpretable Clinical Practice Guidelines and Their Interrelations. In J. Mantas, A. Hasman, P. Gallos, A. Kolokathi, M. S. Househ, & J. Liaskos (Eds.), *Health Informatics Vision: From Data via Information to Knowledge, ICIMTH 2019, 17th International Conference on Informatics, Management and Technology in Healthcare, Athens, Greece, 5-7 July 2019* (Vol. 262, pp. 7–10). IOS Press. https://doi.org/10.3233/SHTI190003

Bottrighi, A., Giordano, L., Molino, G., Montani, S., Terenziani, P., & Torchio, M. (2010). Adopting model checking techniques for clinical guidelines verification. *Artificial Intelligence in Medicine*, *48*(1), 1–19. https://doi.org/10.1016/j.artmed.2009.09.003

Bottrighi, A., Piovesan, L., & Terenziani, P. (2019). Supporting the distributed execution of clinical guidelines by multiple agents. *Artif. Intell. Medicine*, *98*, 87–108. https://doi.org/10.1016/j.artmed.2019.05.001

Brusoni, V., Console, L., Terenziani, P., & Pernid, B. (1997). Later: Managing temporal information efficiently. *IEEE Expert*, *12*(4), 56–64. https://doi.org/10.1109/64.608197

Burghaus, R., Coboeken, K., Gaub, T., Niederalt, C., Sensse, A., Siegmund, H.-U., Weiss, W., Mueck, W., Tanigawa, T., & Lippert, J. (2014). Computational investigation of potential dosing schedules for a switch of medication from warfarin to rivaroxabanâ€"an oral, direct Factor Xa inhibitor. *Frontiers in Physiology*, *5*. https://doi.org/10.3389/fphys.2014.00417

Cairo, M., Hunsberger, L., & Rizzi, R. (2018). Faster Dynamic Controllability Checking for Simple Temporal Networks with Uncertainty. In N. Alechina, K. Nørv\aag, & W. Penczek (Eds.), *25th International Symposium on Temporal Representation and Reasoning (TIME 2018)* (Vol. 120, p. 8:1-8:16). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. https://doi.org/10.4230/LIPIcs.TIME.2018.8

Coles, A. J., Coles, A., Fox, M., & Long, D. (2010). Forward-Chaining Partial-Order Planning. In R. I. Brafman, H. Geffner, J. Hoffmann, & H. A. Kautz (Eds.), *Proceedings of the 20th International Conference on Automated Planning and Scheduling, ICAPS 2010, Toronto, Ontario, Canada, May 12-16, 2010* (pp. 42–49). AAAI. http://www.aaai.org/ocs/index.php/ICAPS/ICAPS10/paper/view/1421

Combi, C., Oliboni, B., & Gabrieli, A. (2015). Conceptual Modeling of Clinical Pathways: Making Data and Processes Connected. In J. H. Holmes, R. Bellazzi, L. Sacchi, & N. Peek (Eds.), *Artificial Intelligence in Medicine – 15th Conference on Artificial Intelligence in Medicine, AIME 2015, Pavia, Italy, June 17-20, 2015. Proceedings* (Vol. 9105, pp. 57–62). Springer. https://doi.org/10.1007/978-3-319-19551-3_7

Combi, C., & Posenato, R. (2018). Extending Conditional Simple Temporal Networks with Partially Shrinkable Uncertainty. In N. Alechina, K. Nørv\aag, & W. Penczek (Eds.), *25th International Symposium on Temporal Representation and Reasoning (TIME 2018)* (Vol. 120, p. 9:1-9:16). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. https://doi.org/10.4230/LIPIcs.TIME.2018.9

Combi, C., Posenato, R., Viganò, L., & Zavatteri, M. (2019). Conditional Simple Temporal Networks with Uncertainty and Resources. *Journal of Artificial Intelligence Research*, *64*, 931-985-931–985. https://doi.org/10.1613/jair.1.11453

Dechter, R., Meiri, I., & Pearl, J. (1991). Temporal Constraint Networks. *Artif. Intell.*, *49*(1–3), 61–95. https://doi.org/10.1016/0004-3702(91)90006-6

Derendorf, H., & Meibohm, B. (1999). Modeling of pharmacokinetic/pharmacodynamic (PK/PD) relationships: Concepts and perspectives. *Pharmaceutical Research*, *16*(2), 176–185. https://doi.org/10.1023/a:1011907920641

Ding, J., Tarokh, V., & Yang, Y. (2018). Model Selection Techniques: An Overview. *IEEE Signal Processing Magazine*, *35*(6), 16–34. https://doi.org/10.1109/MSP.2018.2867638

Dubois, D., Fargier, H., & Prade, H. (1996). Possibility theory in constraint satisfaction problems: Handling priority, preference and uncertainty. *Applied Intelligence*, *6*(4), 287–309. https://doi.org/10.1007/BF00132735

Duftschmid, G., Miksch, S., & Gall, W. (2002). Verification of temporal scheduling constraints in clinical practice guidelines. *Artificial Intelligence in Medicine*, *25*(2), 93–121. https://doi.org/10.1016/S0933-3657(02)00011-8

Ekins, S., Mestres, J., & Testa, B. (2007). In silico pharmacology for drug discovery: Methods for virtual ligand screening and profiling. *British Journal of Pharmacology*, *152*(1), 9–20. https://doi.org/10.1038/sj.bjp.0707305

Emmert-Streib, F., & Dehmer, M. (2019). Evaluation of Regression Models: Model Assessment, Model Selection and Generalization Error. *Machine Learning and Knowledge Extraction*, *1*(1), 521–551. https://doi.org/10.3390/make1010032

Fox, M., & Long, D. (2003). PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. *Journal of Artificial Intelligence Research*, *20*, 61–124. https://doi.org/10.1613/jair.1129

Fraccaro, P., Arguello Castelerio, M., Ainsworth, J., & Buchan, I. (2015). Adoption of Clinical Decision Support in Multimorbidity: A Systematic Review. *JMIR Medical Informatics*, *3*(1), e4. https://doi.org/10.2196/medinform.3503

Gao, M., Popowski, L., & Boerkoel, J. (2020). Dynamic Control of Probabilistic Simple Temporal Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(06), 9851–9858. https://doi.org/10.1609/aaai.v34i06.6538

Guidelines International Network. (n.d.). Guidelines International Network Website. Retrieved October 14, 2014, from http://www.g-i-n.net/

Hunsberger, L., & Posenato, R. (2018). Sound-and-Complete Algorithms for Checking the Dynamic Controllability of Conditional Simple Temporal Networks with Uncertainty. In N. Alechina, K. Nørv\aag, & W. Penczek (Eds.), *25th International Symposium on Temporal Representation and Reasoning (TIME 2018)* (Vol. 120, p. 14:1-14:17). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. https://doi.org/10.4230/LIPIcs.TIME.2018.14

Hunsberger, L., Posenato, R., & Combi, C. (2015). A Sound-and-Complete Propagation-Based Algorithm for Checking the Dynamic Consistency of Conditional Simple Temporal Networks. *2015 22nd International Symposium on Temporal Representation and Reasoning (TIME)*, 4–18. https://doi.org/10.1109/TIME.2015.26

Institute of Medicine, Committee on Quality Health Care in America. (2001). *Crossing the Quality Chasm: A New Health System for the 21st Century*. Washington, USA: National Academy Press. https://www.iom.edu:443/Reports/2001/Crossing-the-Quality-Chasm-A-New-Health-System-for-the-21st-Century.aspx

International Health Terminology Standards Development Organisation. (2015). *SNOMED Clinical Terms*. http://www.ihtsdo.org/snomed-ct

Jafarpour, B., Abidi, S. R., Woensel, W. V., & Abidi, S. S. R. (2019). Execution-time integration of clinical practice guidelines to provide decision support for comorbid conditions. *Artificial Intelligence in Medicine*, *94*, 117–137. https://doi.org/10.1016/j.artmed.2019.02.003

Jafarpour, B., & Abidi, S. S. R. (2013). Merging Disease-Specific Clinical Guidelines to Handle Comorbidities in a Clinical Decision Support Setting. *Artificial Intelligence in Medicine*, 28–32.

Johnstone, R. H., Bardenet, R., Gavaghan, D. J., & Mirams, G. R. (2017). Hierarchical Bayesian inference for ion channel screening dose-response data. *Wellcome Open Research*, *1*, 6. https://doi.org/10.12688/wellcomeopenres.9945.2

Kamisalic, A., Riaño, D., & Welzer, T. (2018). Formalization and acquisition of temporal knowledge for decision support in medical processes. *Comput. Methods Programs Biomed.*, *158*, 207–228. https://doi.org/10.1016/j.cmpb.2018.02.012

Kautz, H. A., & Ladkin, P. B. (1991). Integrating Metric and Qualitative Temporal Reasoning. *Proceedings of the Ninth National Conference on Artificial Intelligence - Volume 1*, 241–246. http://dl.acm.org/citation.cfm?id=1865675.1865713

Khatib, L., Morris, P., Morris, R., & Rossi, F. (2001). Temporal Constraint Reasoning with Preferences. *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 1*, 322–327. http://dl.acm.org/citation.cfm?id=1642090.1642135

Kogan, A., Peleg, M., Tu, S. W., Allon, R., Khaitov, N., & Hochberg, I. (2020). Towards a goal-oriented methodology for clinical-guideline-based management recommendations for patients with multimorbidity: GoCom and its preliminary evaluation. *Journal of Biomedical Informatics*, *112*, 103587.

https://doi.org/10.1016/j.jbi.2020.103587
Kogan, A., Tu, S. W., & Peleg, M. (2018). Goal-driven management of interacting clinical guidelines for multi-morbidity patients. *AMIA 2018, American Medical Informatics Association Annual Symposium, San Francisco, CA, November 3-7, 2018*. http://knowledge.amia.org/67852-amia-1.4259402/t004-1.4263758/t004-1.4263759/2977329-1.4263913/2969812-1.4263910
Koubarakis, M. (1997). From local to global consistency in temporal constraint networks. *Theoretical Computer Science*, *173*(1), 89–112. https://doi.org/10.1016/S0304-3975(96)00192-2
Lin Xu, & Choueiry, B. Y. (2003). A new effcient algorithm for solving the simple temporal problem. *10th International Symposium on Temporal Representation and Reasoning, 2003 and Fourth International Conference on Temporal Logic. Proceedings.*, 210–220. https://doi.org/10.1109/TIME.2003.1214898
Mehrotra, N., Gupta, M., Kovar, A., & Meibohm, B. (2007). The role of pharmacokinetics and pharmacodynamics in phosphodiesterase-5 inhibitor therapy. *International Journal of Impotence Research*, *19*(3), 253–264. https://doi.org/10.1038/sj.ijir.3901522
Meiri, I. (1996). Combining Qualitative and Quantitative Constraints in Temporal Reasoning. *Artif. Intell.*, *87*(1–2), 343–385. https://doi.org/10.1016/0004-3702(95)00109-3
Merhej, E., Schockaert, S., McKelvey, T. G., & De Cock, M. (2016). Generating conflict-free treatments for patients with comorbidity using ASP. In *KR4HC 2016* (pp. 93--100).
Michalowski, M., Wilk, S., Michalowski, W., & Carrier, M. (2021). MitPlan: A planning approach to mitigating concurrently applied clinical practice guidelines. *Artificial Intelligence in Medicine*, *112*. Scopus. https://doi.org/10.1016/j.artmed.2020.102002
Montani, S., & Terenziani, P. (2006). Exploiting decision theory concepts within clinical guideline systems: Toward a general approach. *Int. J. Intell. Syst.*, *21*(6), 585–599. https://doi.org/10.1002/int.20149
Morris, P., Muscettola, N., & Vidal, T. (2001). Dynamic control of plans with temporal uncertainty. *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 1*, 494–499.
Mouhoub, M., & Liu, J. (2008). Managing uncertain temporal relations using a probabilistic Interval Algebra. *2008 IEEE International Conference on Systems, Man and Cybernetics*, 3399–3404. https://doi.org/10.1109/ICSMC.2008.4811823
Mouhoub, M., & Sukpan, A. (2008). Managing Temporal Constraints with Preferences. *Spatial Cognition & Computation*, *8*(1–2), 131–149. https://doi.org/10.1080/13875860801930407
Musen, M. A. (2015). The Protégé Project: A Look Back and a Look Forward. *AI Matters*, *1*(4), 4–12. https://doi.org/10.1145/2757001.2757003
NICE | The National Institute for Health and Care Excellence. (n.d.). [CorporatePage]. NICE; NICE. Retrieved May 29, 2020, from https://www.nice.org.uk/
Peleg, M. (2013). Computer-interpretable clinical guidelines: A methodological review. *Journal of Biomedical Informatics*, *46*(4), 744–763. https://doi.org/10.1016/j.jbi.2013.06.009
Piovesan, L., Molino, G., & Terenziani, P. (2014). Supporting Physicians in the Detection of the Interactions between Treatments of Co-Morbid Patients. In *Healthcare Informatics and Analytics: Emerging Issues and Trends* (pp. 165–193). IGI Global.
Piovesan, L., Molino, G., & Terenziani, P. (2015). Supporting Multi-Level User-Driven Detection of Guideline Interactions. *Proceedings of the International Conference on Health Informatics (HEALTHINF-2015)*, 413–422. https://doi.org/10.5220/0005217404130422
Piovesan, L., & Terenziani, P. (2015). A Mixed-Initiative approach to the conciliation of Clinical Guidelines for comorbid patients. In *KR4HC 2015* (Vol. 9485, pp. 95–108). Springer International Publishing.
Piovesan, L., & Terenziani, P. (2016). A Constraint-Based Approach for the Conciliation of Clinical Guidelines. *Advances in Artificial Intelligence - IBERAMIA 2016*, *10022*, 77–88. https://doi.org/10.1007/978-3-319-47955-2_7
Piovesan, L., Terenziani, P., & Molino, G. (2018). GLARE-SSCPM: An Intelligent System to Support the Treatment of Comorbid Patients. *IEEE Intelligent Systems*. https://doi.org/10.1109/MIS.2018.111144734
Piovesan, L., Terenziani, P., & Theseider Dupré, D. (2020). Conformance analysis for comorbid patients in Answer Set Programming. *Journal of Biomedical Informatics*, *103*, 103377. https://doi.org/10.1016/j.jbi.2020.103377
Planken, L., de Weerdt, M., & van der Krogt, R. (2008). P3C: A New Algorithm for the Simple Temporal Problem. *Proceedings of the Eighteenth International Conference on International Conference on Automated Planning and Scheduling*, 256–263. http://dl.acm.org/citation.cfm?id=3037281.3037314
Riaño, D., & Collado, A. (2013). Model-Based Combination of Treatments for the Management of Chronic Comorbid Patients. In *Artificial Intelligence in Medicine* (Vol. 7885, pp. 11–16). Springer Berlin Heidelberg. http://link.springer.com/10.1007/978-3-642-38326-7_2
Riaño, D., & Ortega, W. (2017). Computer technologies to integrate medical treatments to manage multimorbidity. *Journal of Biomedical Informatics*, *75*, 1–13. https://doi.org/10.1016/j.jbi.2017.09.009
Ryabov, V., & Trudel, A. (2004). Probabilistic temporal interval networks. *Proceedings. 11th International Symposium on Temporal Representation and Reasoning, 2004. TIME 2004.*, 64–67. https://doi.org/10.1109/TIME.2004.1314421

Sánchez-Garzón, I., Fdez-Olivares, J., Onaindía, E., Milla, G., Jordán, J., & Castejón, P. (2013). A Multi-agent Planning Approach for the Generation of Personalized Treatment Plans of Comorbid Patients. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, & G. Weikum (Eds.), *AIME 2013* (Vol. 7885, pp. 23–27). Springer Berlin Heidelberg. http://link.springer.com/10.1007/978-3-642-38326-7_4

Schwalb, E., & Vila, L. (1998). Temporal Constraints: A Survey. *Constraints*, *3*(2–3), 129–149. https://doi.org/10.1023/A:1009717525330

Shahar, Y., Miksch, S., & Johnson, P. (1998). The Asgaard project: A task-specific framework for the application and critiquing of time-oriented clinical guidelines. *Artificial Intelligence in Medicine*, *14*(1–2), 29–51.

Sittig, D. F., Wright, A., Osheroff, J. A., Middleton, B., Teich, J. M., Ash, J. S., Campbell, E., & Bates, D. W. (2008). Grand challenges in clinical decision support. *Journal of Biomedical Informatics*, *41*(2), 387–392. https://doi.org/10.1016/j.jbi.2007.09.003

Spiotta, M., Terenziani, P., & Dupré, D. T. (2017). Temporal Conformance Analysis and Explanation of Clinical Guidelines Execution: An Answer Set Programming Approach. *IEEE Trans. Knowl. Data Eng.*, *29*(11), 2567–2580. https://doi.org/10.1109/TKDE.2017.2734084

Spruill, W. J., Wade, W. E., DiPiro, J. T., Blouin, R. A., & Pruemer, J. M. (2014). *Concepts in clinical pharmacokinetics* (Sixth edition). American Society of Health-System Pharmacists.

Ten Teije, A., Miksch, S., & Lucas, P. (Eds.). (2008). *Computer-based medical guidelines and protocols: A primer and current trends* (Vol. 139). IOS Press. http://dl.acm.org/citation.cfm?id=1479604

Terenziani, P. (2006). Reasoning about Time. In *Encyclopedia of Cognitive Science* (pp. 869–874). John Wiley & Sons, Ltd. http://onlinelibrary.wiley.com/doi/10.1002/0470018860.s00089/abstract

Terenziani, P., & Andolina, A. (2019). Considering Temporal Preferences and Probabilities in Guideline Interaction Analysis. In D. Riaño, S. Wilk, & A. ten Teije (Eds.), *Artificial Intelligence in Medicine – 17th Conference on Artificial Intelligence in Medicine, AIME 2019, Poznan, Poland, June 26-29, 2019, Proceedings* (Vol. 11526, pp. 120–124). Springer. https://doi.org/10.1007/978-3-030-21642-9_16

Terenziani, P., & Andolina, A. (2017). Probabilistic quantitative temporal reasoning. In A. Seffah, B. Penzenstadler, C. Alves, & X. Peng (Eds.), *Proceedings of the Symposium on Applied Computing, SAC 2017, Marrakech, Morocco, April 3-7, 2017* (pp. 965–970). ACM. https://doi.org/10.1145/3019612.3019712

Terenziani, P., Andolina, A., & Piovesan, L. (2017). Managing Temporal Constraints with Preferences: Representation, Reasoning, and Querying. *IEEE Transactions on Knowledge and Data Engineering*, *29*(9), 2067–2071. https://doi.org/10.1109/TKDE.2017.2697852

Terenziani, P., German, E., & Shahar, Y. (2008). The temporal aspects of clinical guidelines. *Studies in Health Technology and Informatics*, *139*, 81–100.

Terenziani, P., Molino, G., & Torchio, M. (2001). A modular approach for representing and executing clinical guidelines. *Artificial Intelligence in Medicine*, *23*(3), 249–276. https://doi.org/10.1016/S0933-3657(01)00087-2

Terenziani, P., Montani, S., Bottrighi, A., Torchio, M., Molino, G., & Correndo, G. (2004). A context-adaptable approach to clinical guidelines. *Studies in Health Technology and Informatics*, *107*(Pt 1), 169–173.

Van Woensel, W., Abidi, S. S. R., & Abidi, S. R. (2021). Decision support for comorbid conditions via execution-time integration of clinical guidelines using transaction-based semantics and temporal planning. *Artificial Intelligence in Medicine*, *118*. Scopus. https://doi.org/10.1016/j.artmed.2021.102127

Vila, L. (1994). A Survey on Temporal Reasoning in Artificial Intelligence. *AI Commun.*, *7*(1), 4–28.

Vilain, M. (1982). A System for Reasoning About Time. In D. L. Waltz (Ed.), *Proceedings of the National Conference on Artificial Intelligence. Pittsburgh, PA, August 18-20, 1982* (pp. 197–201). AAAI Press.

Vilain, M. B., & Kautz, H. A. (1986). Constraint Propagation Algorithms for Temporal Reasoning. In T. Kehler (Ed.), *Proceedings of the 5th National Conference on Artificial Intelligence. Philadelphia, PA, August 11-15, 1986. Volume 1: Science* (pp. 377–382). Morgan Kaufmann. http://www.aaai.org/Library/AAAI/1986/aaai86-063.php

WHO Collaborating Centre for Drug Statistics Methodology. (n.d.). *Anatomical Therapeutic Chemical classification system*. Retrieved October 14, 2014, from http://www.whocc.no/atc/

Wilk, S., Michalowski, M., Michalowski, W., Rosu, D., Carrier, M., & Kezadri-Hamiaz, M. (2017). Comprehensive mitigation framework for concurrent application of multiple clinical practice guidelines. *Journal of Biomedical Informatics*, *66*, 52–71. https://doi.org/10.1016/j.jbi.2016.12.002

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., Assempour, N., Iynkkaran, I., Liu, Y., Maciejewski, A., Gale, N., Wilson, A., Chin, L., Cummings, R., Le, D., … Wilson, M. (2018). DrugBank 5.0: A major update to the DrugBank database for 2018. *Nucleic Acids Research*, *46*(D1), D1074–D1082. https://doi.org/10.1093/nar/gkx1037

Zamborlini, V., da Silveira, M., Pruski, C., ten Teije, A., Geleijn, E., van der Leeden, M., Stuiver, M., & van Harmelen, F. (2017). Analyzing interactions on combining multiple clinical guidelines. *Artificial Intelligence in Medicine*. https://doi.org/10.1016/j.artmed.2017.03.012

Zavatteri, M., Combi, C., Rizzi, R., & Viganò, L. (2019). Hybrid SAT-Based Consistency Checking Algorithms for

Simple Temporal Networks with Decisions. In J. Gamper, S. Pinchinat, & G. Sciavicco (Eds.), *26th International Symposium on Temporal Representation and Reasoning (TIME 2019)* (Vol. 147, p. 16:1-16:17). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. https://doi.org/10.4230/LIPIcs.TIME.2019.16

Zavatteri, M., & Viganò, L. (2019). Conditional simple temporal networks with uncertainty and decisions. *Theoretical Computer Science*, *797*, 77–101. https://doi.org/10.1016/j.tcs.2018.09.023

Zhang, Y., & Zhang, Z. (2014). Preliminary Result on Finding Treatments for Patients with Comorbidity. In S. Miksch, D. Riaño, & A. ten Teije (Eds.), *Knowledge Representation for Health Care* (pp. 14–28). Springer International Publishing. https://doi.org/10.1007/978-3-319-13281-5_2

APPENDIX A

In this section, we present in Fig. 11 the minimal network obtained from applying our temporal reasoning algorithm to the graph of Fig. 7 (representing the constraints in Ex.1+Ex.1'). For the sake of readability, in the graph of Fig. 11, the

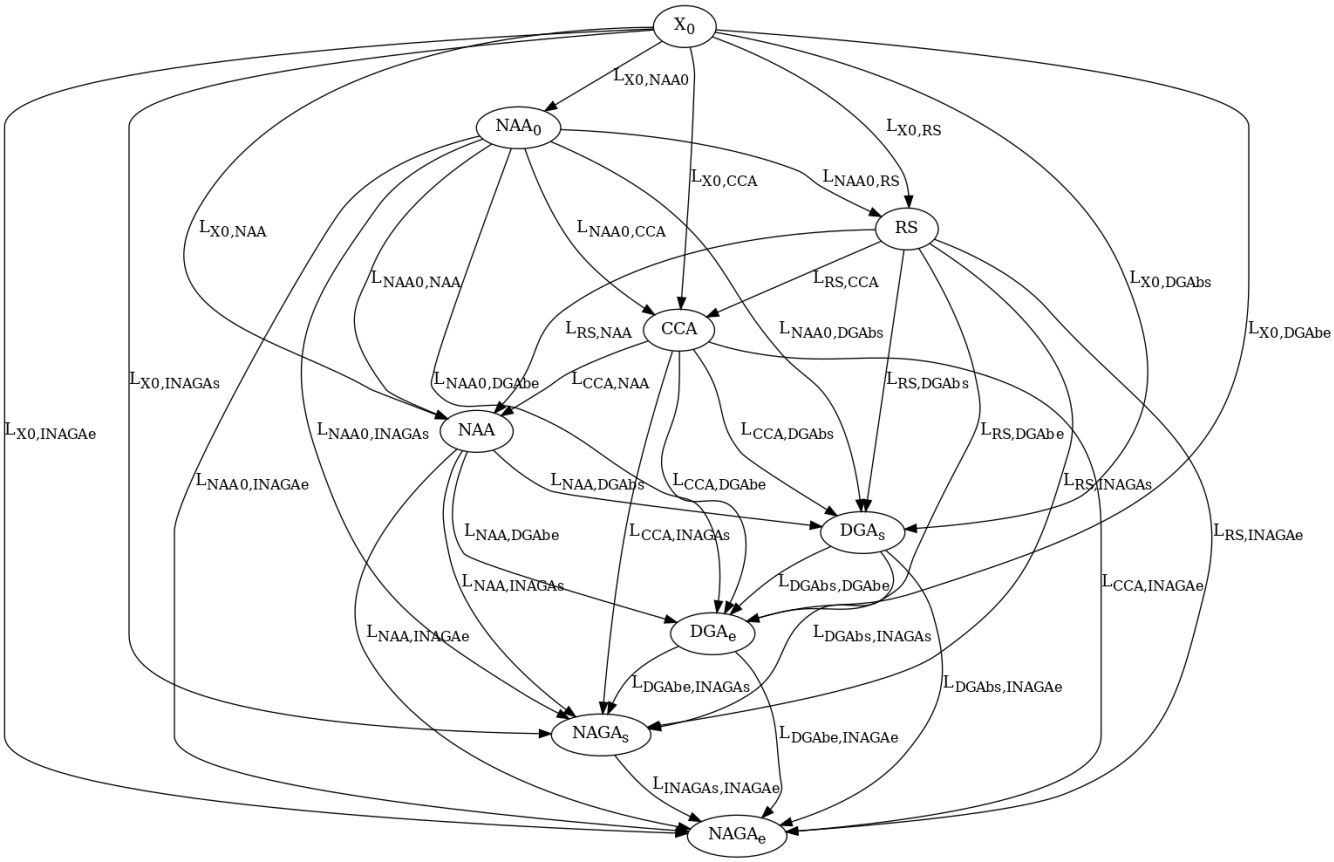

Fig. 11 Minimal network resulting from the application of our temporal reasoning algorithm to the graph of Fig. 7.

label of each edge (i, j) has been replaced by the symbol $L_{i,j}$, where i and j are two vertices of the graph, and the corresponding P+PQTL (i.e., the label of constraint $iL_{i,j}j$) is reported separately in Table 2, where column "Edge Label" contains an edge label as found in Fig. 7, and column "P+PQTL" contains the P+PQTL associated with the edge label of the same row. For instance, the "Edge Label" $L_{X_0,CCA}$ represents the label of edge (X_0, CCA) and its associated P+PQTL is <(52, 0.0134607, 1), (53, 0.0813113, 1), (54, 0.218911, 0.75), (55, 0.311709, 0.75), (56, 0.251776, 0.5), (57, 0.104329, 0.5), (58, 0.018503, 0.25)>.

Table 2. Edge labels (i.e., P+PQTLs) for the minimal network of Fig. 7.

Edge Label	P+PQTL
L_{X_0,NAA_0}	<(28, 1, 1)>
$L_{X_0,RS}$	<(52, 1, 1)>
$L_{X_0,CCA}$	<(52, 0.0134607, 1), (53, 0.0813113, 1), (54, 0.218911, 0.75), (55, 0.311709, 0.75), (56, 0.251776, 0.5), (57, 0.104329, 0.5), (58, 0.018503, 0.25)>
$L_{X_0,NAA}$	<(48, 0.00142083, 0.25), (49, 0.0154645, 0.5), (50, 0.0782606, 0.75), (51, 0.210637, 1), (52, 0.309997, 1), (53, 0.249903, 1), (54, 0.108162, 0.75), (55, 0.0239305, 0.5), (56, 0.00222469, 0.25)>
$L_{X_0,DGAb_s}$	<(53, 0.00402692, 1), (54, 0.0367196, 1), (55, 0.130377, 1), (56, 0.235721, 1), (57, 0.274475, 0.75), (58, 0.205071, 0.75), (59, 0.0962546, 0.5), (60, 0.0164411, 0.5), (61, 0.000914591, 0.25)>
$L_{X_0,DGAb_e}$	<(57, 9.51058e-05, 1), (58, 0.00236306, 1), (59, 0.0220704, 1), (60, 0.0844066, 1), (61, 0.180308, 1), (62,

	0.265044, 1), (63, 0.247121, 1), (64, 0.142905, 0.75), (65, 0.0475228, 0.75), (66, 0.00747661, 0.5), (67, 0.000652556, 0.5), (68, 3.48505e-05, 0.25)>
$L_{X_0,INAGA_s}$	<(49, 0.00203797, 0.25), (50, 0.0201031, 0.5), (51, 0.0642004, 0.75), (52, 0.143918, 1), (53, 0.224908, 1), (54, 0.244054, 1), (55, 0.180752, 1), (56, 0.0887179, 0.75), (57, 0.0277925, 0.5), (58, 0.00351621, 0.25)>
$L_{X_0,,INAGA_e}$	<(50, 8.75086e-05, 0.25), (51, 0.00184127, 0.5), (52, 0.0133795, 0.75), (53, 0.0493056, 1), (54, 0.117447, 1), (55, 0.192798, 1), (56, 0.231953, 1), (57, 0.199543, 1), (58, 0.125901, 1), (59, 0.0517802, 1), (60, 0.0134681, 1), (61, 0.00221205, 1), (62, 0.000261647, 0.75), (63, 1.95837e-05, 0.5), (64, 8.76022e-07, 0.25)>
$L_{NAA_0,RS}$	<(24, 1, 1)>
$L_{NAA_0,CCA}$	<(24, 0.0134607, 1), (25, 0.0813113, 1), (26, 0.218911, 0.75), (27, 0.311709, 0.75), (28, 0.251776, 0.5), (29, 0.104329, 0.5), (30, 0.018503, 0.25)>
$L_{NAA_0,NAA}$	<(20, 0.00142083, 0.25), (21, 0.0154645, 0.5), (22, 0.0782606, 0.75), (23, 0.210637, 1), (24, 0.309997, 1), (25, 0.249903, 1), (26, 0.108162, 0.75), (27, 0.0239305, 0.5), (28, 0.00222469, 0.25)>
$L_{NAA_0,DGAb_s}$	<(25, 0.00402692, 1), (26, 0.0367196, 1), (27, 0.130377, 1), (28, 0.235721, 1), (29, 0.274475, 0.75), (30, 0.205071, 0.75), (31, 0.0962546, 0.5), (32, 0.0164411, 0.5), (33, 0.000914591, 0.25)>
$L_{NAA_0,DGAb_e}$	<(29, 9.51058e-05, 1), (30, 0.00236306, 1), (31, 0.0220704, 1), (32, 0.0844066, 1), (33, 0.180308, 1), (34, 0.265044, 1), (35, 0.247121, 1), (36, 0.142905, 0.75), (37, 0.0475228, 0.75), (38, 0.00747661, 0.5), (39, 0.000652556, 0.5), (40, 3.48505e-05, 0.25)>
$L_{NAA_0,INAGA_s}$	<(21, 0.00203797, 0.25), (22, 0.0201031, 0.5), (23, 0.0642004, 0.75), (24, 0.143918, 1), (25, 0.224908, 1), (26, 0.244054, 1), (27, 0.180752, 1), (28, 0.0887179, 0.75), (29, 0.0277925, 0.5), (30, 0.00351621, 0.25)>
$L_{NAA_0,INAGA_e}$	<(22, 8.75086e-05, 0.25), (23, 0.00184127, 0.5), (24, 0.0133795, 0.75), (25, 0.0493056, 1), (26, 0.117447, 1), (27, 0.192798, 1), (28, 0.231953, 1), (29, 0.199543, 1), (30, 0.125901, 1), (31, 0.0517802, 1), (32, 0.0134681, 1), (33, 0.00221205, 1), (34, 0.000261647, 0.75), (35, 1.95837e-05, 0.5), (36, 8.76022e-07, 0.25)>
$L_{RS,CCA}$	<(0, 0.0134607, 1), (1, 0.0813113, 1), (2, 0.218911, 0.75), (3, 0.311709, 0.75), (4, 0.251776, 0.5), (5, 0.104329, 0.5), (6, 0.018503, 0.25)>
$L_{RS,NAA}$	<(-4, 0.00142083, 0.25), (-3, 0.0154645, 0.5), (-2, 0.0782606, 0.75), (-1, 0.210637, 1), (0, 0.309997, 1), (1, 0.249903, 1), (2, 0.108162, 0.75), (3, 0.0239305, 0.5), (4, 0.00222469, 0.25)>
$L_{RS,DGAb_s}$	<(1, 0.00402692, 1), (2, 0.0367196, 1), (3, 0.130377, 1), (4, 0.235721, 1), (5, 0.274475, 0.75), (6, 0.205071, 0.75), (7, 0.0962546, 0.5), (8, 0.0164411, 0.5), (9, 0.000914591, 0.25)>
$L_{RS,DGAb_e}$	<(5, 9.51058e-05, 1), (6, 0.00236306, 1), (7, 0.0220704, 1), (8, 0.0844066, 1), (9, 0.180308, 1), (10, 0.265044, 1), (11, 0.247121, 1), (12, 0.142905, 0.75), (13, 0.0475228, 0.75), (14, 0.00747661, 0.5), (15, 0.000652556, 0.5), (16, 3.48505e-05, 0.25)>
$L_{RS,INAGA_s}$	<(-3, 0.00203797, 0.25), (-2, 0.0201031, 0.5), (-1, 0.0642004, 0.75), (0, 0.143918, 1), (1, 0.224908, 1), (2, 0.244054, 1), (3, 0.180752, 1), (4, 0.0887179, 0.75), (5, 0.0277925, 0.5), (6, 0.00351621, 0.25)>
$L_{RS,INAGA_e}$	<(-2, 8.75086e-05, 0.25), (-1, 0.00184127, 0.5), (0, 0.0133795, 0.75), (1, 0.0493056, 1), (2, 0.117447, 1), (3, 0.192798, 1), (4, 0.231953, 1), (5, 0.199543, 1), (6, 0.125901, 1), (7, 0.0517802, 1), (8, 0.0134681, 1), (9, 0.00221205, 1), (10, 0.000261647, 0.75), (11, 1.95837e-05, 0.5), (12, 8.76022e-07, 0.25)>
$L_{CCA,NAA}$	<(-10, 2.79754e-09, 0.25), (-9, 6.33977e-07, 0.25), (-8, 3.63775e-05, 0.5), (-7, 0.000868524, 0.5), (-6, 0.0105809, 0.5), (-5, 0.069516, 0.75), (-4, 0.235043, 0.75), (-3, 0.341664, 0.75), (-2, 0.250559, 1), (-1, 0.0784016, 1), (0, 0.0123006, 1), (1, 0.00099186, 1), (2, 3.78892e-05, 0.75), (3, 5.50532e-07, 0.5), (4, 1.85569e-09, 0.25)>
$L_{CCA,DGAb_s}$	<(1, 0.395458, 1), (2, 0.456637, 1), (3, 0.147905, 1)>
$L_{CCA,DGAb_e}$	<(5, 0.0373862, 1), (6, 0.204006, 1), (7, 0.444584, 1), (8, 0.239003, 1), (9, 0.0575214, 1), (10, 0.0175001,

	1)>
$L_{CCA,INAGA_e}$	<(-9, 3.21271e-07, 0.25), (-8, 2.2304e-05, 0.25), (-7, 0.000375183, 0.5), (-6, 0.0031125, 0.5), (-5, 0.0161265, 0.5), (-4, 0.0580381, 0.75), (-3, 0.148034, 0.75), (-2, 0.247665, 0.75), (-1, 0.26068, 1), (0, 0.174456, 1), (1, 0.0700768, 1), (2, 0.0181684, 1), (3, 0.00295316, 1), (4, 0.000278229, 0.75), (5, 1.26466e-05, 0.5), (6, 1.68126e-07, 0.25)>
$L_{CCA,INAGA_e}$	<(-8, 1.53175e-09, 0.25), (-7, 2.9084e-07, 0.25), (-6, 1.28157e-05, 0.5), (-5, 0.000224054, 0.5), (-4, 0.00209349, 0.5), (-3, 0.0121372, 0.75), (-2, 0.0470996, 0.75), (-1, 0.121247, 0.75), (0, 0.209745, 1), (1, 0.244017, 1), (2, 0.194902, 1), (3, 0.109041, 1), (4, 0.043944, 1), (5, 0.0126447, 1), (6, 0.00253928, 1), (7, 0.000325682, 1), (8, 2.57906e-05, 1), (9, 1.23706e-06, 1), (10, 3.59029e-08, 0.75), (11, 5.25712e-10, 0.5), (12, 2.80375e-12, 0.25)>
$L_{NAA,DGAb_s}$	<(-3, 9.25544e-09, 0.25), (-2, 1.36158e-06, 0.5), (-1, 5.3611e-05, 0.75), (0, 0.000886938, 1), (1, 0.00785679, 1), (2, 0.0414772, 1), (3, 0.134883, 1), (4, 0.255612, 1), (5, 0.286393, 1), (6, 0.184392, 0.75), (7, 0.0699182, 0.75), (8, 0.0160397, 0.75), (9, 0.0022845, 0.5), (10, 0.000192658, 0.5), (11, 8.38491e-06, 0.5), (12, 1.28364e-07, 0.25), (13, 4.57248e-10, 0.25)>
$L_{NAA,DGAb_e}$	<(1, 9.66332e-11, 0.25), (2, 3.67129e-08, 0.5), (3, 3.05891e-06, 0.75), (4, 8.28895e-05, 1), (5, 0.00103093, 1), (6, 0.00738966, 1), (7, 0.0333132, 1), (8, 0.0988267, 1), (9, 0.195765, 1), (10, 0.254491, 1), (11, 0.219501, 1), (12, 0.125849, 1), (13, 0.0484897, 0.75), (14, 0.0128175, 0.75), (15, 0.00220078, 0.75), (16, 0.000226785, 0.5), (17, 1.24905e-05, 0.5), (18, 3.17585e-07, 0.5), (19, 3.65858e-09, 0.25), (20, 1.54687e-11, 0.25)>
$L_{NAA,INAGA_s}$	<(1, 0.389832, 1), (2, 0.610168, 1)>
$L_{NAA,INAGA_e}$	<(2, 0.0931263, 1), (3, 0.330536, 1), (4, 0.367839, 1), (5, 0.148214, 1), (6, 0.0476238, 1), (7, 0.00944498, 1), (8, 0.00321607, 1)>
$L_{DGAb_s,DGAb_e}$	<(2, 0.00661778, 1), (3, 0.0513521, 1), (4, 0.179003, 1), (5, 0.29579, 1), (6, 0.269049, 1), (7, 0.136086, 1), (8, 0.0495023, 1), (9, 0.0126001, 1)>
$L_{DGAb_s,INAGA_s}$	<(-12, 1.10228e-07, 0.25), (-11, 6.1068e-06, 0.25), (-10, 0.000114232, 0.5), (-9, 0.00103306, 0.5), (-8, 0.00558142, 0.5), (-7, 0.0211963, 0.75), (-6, 0.0601589, 0.75), (-5, 0.128671, 0.75), (-4, 0.202865, 1), (-3, 0.23002, 1), (-2, 0.185483, 1), (-1, 0.106282, 1), (0, 0.0431397, 1), (1, 0.0125767, 1), (2, 0.00252839, 1), (3, 0.000321338, 0.75), (4, 2.1753e-05, 0.5), (5, 5.49598e-07, 0.25)>
$L_{DGAb_s,INAGA_e}$	<(-11, 1.60081e-09, 0.25), (-10, 2.12132e-07, 0.25), (-9, 7.68125e-06, 0.5), (-8, 0.000116029, 0.5), (-7, 0.000952951, 0.5), (-6, 0.00500682, 0.75), (-5, 0.0186515, 0.75), (-4, 0.0516438, 0.75), (-3, 0.108024, 1), (-2, 0.170696, 1), (-1, 0.204837, 1), (0, 0.187752, 1), (1, 0.13305, 1), (2, 0.0735984, 1), (3, 0.031868, 1), (4, 0.010615, 1), (5, 0.0026415, 1), (6, 0.000472828, 1), (7, 6.02336e-05, 1), (8, 5.38618e-06, 1), (9, 3.22431e-07, 0.75), (10, 1.07253e-08, 0.5), (11, 1.36229e-10, 0.25)>
$L_{DGAb_e,INAGA_s}$	<(-19, 1.11902e-08, 0.25), (-18, 6.10847e-07, 0.25), (-17, 1.21623e-05, 0.5), (-16, 0.000142447, 0.5), (-15, 0.0010472, 0.5), (-14, 0.00509594, 0.75), (-13, 0.0177611, 0.75), (-12, 0.0467918, 0.75), (-11, 0.0966001, 1), (-10, 0.157628, 1), (-9, 0.199916, 1), (-8, 0.19543, 1), (-7, 0.146012, 1), (-6, 0.0829385, 1), (-5, 0.0358663, 1), (-4, 0.0115915, 1), (-3, 0.00270455, 1), (-2, 0.000422238, 1), (-1, 3.77454e-05, 0.75), (0, 1.57458e-06, 0.5), (1, 2.24968e-08, 0.25)>
$L_{DGAb_e,INAGA_e}$	<(-18, 2.75948e-10, 0.25), (-17, 2.95143e-08, 0.25), (-16, 1.06945e-06, 0.5), (-15, 1.92606e-05, 0.5), (-14, 0.000197294, 0.5), (-13, 0.00126094, 0.75), (-12, 0.00547715, 0.75), (-11, 0.0173714, 0.75), (-10, 0.0426131, 1), (-9, 0.0835243, 1), (-8, 0.132477, 1), (-7, 0.170711, 1), (-6, 0.179039, 1), (-5, 0.153341, 1), (-4, 0.107709, 1), (-3, 0.0620746, 1), (-2, 0.0291629, 1), (-1, 0.0109733, 1), (0, 0.00320921, 1), (1, 0.000708433, 1), (2, 0.000115364, 1), (3, 1.363e-05, 1), (4, 1.12365e-06, 1), (5, 5.74542e-08, 0.75), (6, 1.39595e-09, 0.5), (7, 1.08275e-11, 0.25)>
$L_{INAGA_s,INAGA_e}$	<(0, 0.0758523, 1), (1, 0.213296, 1), (2, 0.278516, 1), (3, 0.211482, 1), (4, 0.121602, 1), (5, 0.0588336, 1), (6, 0.0317201, 1), (7, 0.00869848, 1)>