Temporal Detection of Guideline Interactions

Luca Piovesan[1], Luca Anselma[1] and Paolo Terenziani[2]
[1]*Dipartimento di Informatica, Università degli Studi di Torino, Torino, Italy*
[2]*DISIT, Università del Piemonte Orientale "Amedeo Avogadro", Alessandria, Italy*

Keywords: Computer-interpretable Clinical Guidelines, Comorbidity Treatment, Guideline Interaction Detection, Ontology of Time and Interactions, Temporal Reasoning.

Abstract: Clinical practice guidelines are widely used to support physicians, but only on individual pathologies. On the other hand, the treatment of patients affected by multiple diseases is one of the main challenges for the modern healthcare. This requires the development of new methodologies, supporting physicians in the detection of interactions between guidelines. In a previous work, we proposed a flexible and user-driven approach, helping physicians in the detection of possible interactions between guidelines, supporting focusing and analysis at multiple levels of abstractions. However, it did not cope with the fact that interactions occur in time. For instance, the effects of two actions may potentially conflict, but practical conflicts happen only if such effects overlap in time. In this paper, we extend the ontological model to deal with the temporal aspects, and the detection algorithms to cope with them. Different types of facilities are provided to physicians, supporting the analysis of interactions between both guidelines "per se", and the concrete application of guidelines to specific patients. In both cases, different temporal facilities are provided to user physicians, based on Artificial Intelligence temporal reasoning techniques.

1 INTRODUCTION

The research about computer-interpretable clinical guidelines (henceforth CIGs) has gained a relevant role within the Medical Informatics community. In the last twenty years, several different approaches and projects have been developed to create domain-independent computer-assisted tools for managing, acquiring, representing and executing CIGs (consider, e.g., the collections (Gordon and Christensen 1995; Fridsma 2001; Ten Teije et al. 2008; Peleg 2013)).

By definition, clinical guidelines address specific clinical circumstances (i.e., specific diseases). However, unfortunately, specific patients may be affected by more than one disease. The treatment of *comorbid patients* (i.e., patients affected by multiple diseases) is one of the main challenges for the modern health care, also due to the aging of population, and the consequent increase of chronic diseases. This sets up the urgent need of developing ways of merging multiple single-disease interventions to provide professionals' assistance to comorbid patients (Riaño and Collado 2013).

However, though some CIGs covering frequently occurring comorbidities might be devised, the approach of considering all the possible combinations of pathologies does not scale up. Thus, there is a need for formal methodologies to support physicians in the detection and resolution of interactions between guidelines, and, ultimately, in the process of merging two or more guidelines. This is an increasingly "hot topic" within the Medical Informatics community, and several approaches have been proposed in the last years (see Section 5).

In a recent work in this context, we faced a central issue in the management of multiple CIGs, namely the **interaction detection**. In (Piovesan et al. 2014), we identified three different knowledge levels at which interactions might occur: (i) level of the intentions of the CIG actions, (ii) level of the goals of the drug categories (recommended by the pharmaceutical actions in the CIGs), and (iii) level of drugs. We have also pointed out that, in turn, levels (i) and (ii) may be structured at different levels of detail. In (Piovesan et al. 2014), we have also proposed an ontological representation for the interactions at the different levels, as well as support for interactive physician-driven analysis of the interactions, at the different levels.

Nonetheless, to the best of our knowledge, until now no CIG approach in the literature has focused

Piovesan L., Anselma L. and Terenziani P..
Temporal Detection of Guideline Interactions.
DOI: 10.5220/0005186300400050
In *Proceedings of the International Conference on Health Informatics* (HEALTHINF-2015), pages 40-50
ISBN: 978-989-758-068-0

on the temporal aspects of interactions. Indeed, a non-temporal analysis can detect a possible interaction between two actions in different CIGs, identifying, e.g., a potential conflict between their intentions (or effects). However, as long as no temporal analysis is performed, such an interaction is only "hypothetical": actual interactions occur (and the user physician should consider it) only in the case that the conflicting intentions or effects overlap in time. The approach in this paper is, to the best of our knowledge, the first one starting to face such a challenging problem. Indeed, we aim at supporting physicians in the temporal analysis of interactions in both "abstract" analysis of CIGs (not considering patient data) and in the analysis during the execution on specific patients.

In this paper, we consider time information about action execution, effects and goals. After introducing some preliminaries (Section 2), we propose a representation formalism to model such information (Section 3). Unfortunately, the only representation of such knowledge is not enough to support interaction detection: to this purpose, we propose correct and complete temporal constraint propagation techniques (Section 4). In particular, on top of the temporal reasoning engine, we provide users with different temporal facilities, to support different forms of interaction detection. Finally, Section 5 contains related works and conclusions.

2 PRELIMINARIES

Though the methodology we propose in this paper is mostly system-independent, we based our approach on GLARE (Subsection 2.1). In this preliminary section, we also briefly describe our previous work about comorbidity detection. In Subsection 2.2 we summarize an ontology for interactions, and in Subsection 2.3 we mention a (non-temporal) detection algorithm.

2.1 Glare

GLARE (Guideline Acquisition, Representation and Execution) has been built starting from 1997 in a long-term cooperation between the Department of Computer Science of the University of Piemonte Orientale, Alessandria, Italy, and the Azienda Ospedaliera San Giovanni Battista in Turin (one of the largest hospitals in Italy).

GLARE supports the use of advanced artificial intelligence techniques and decision support techniques to assist physicians in merging two or more guidelines developed for the treatment of individual diseases.

In this paper, we extend GLARE to cope with comorbidities. Our goal is twofold: on a side, we aim to build a system able to, during the merging process, draw "intelligent" conclusions starting from the knowledge about CIGs; on the other, the system must be "collaborative". This last desideratum is due to the stance that, when facing decision making in medical informatics, black-box tools that take decisions for her/him could be not very useful for the user physician. Instead, a tool that guides her/him in the decision-making process, helping her/him to integrate also the knowledge that is not modelled in the system but that (s)he owns, is more useful and could improve the quality of the decisions obtained. This is also the underlying philosophy of the *mixed initiative* approach in artificial intelligence and human-computer interaction. In fact, Horvitz (1999) defines mixed initiative as "*methods that explicitly support an efficient, natural interleaving of contributions by users and automated services aimed at converging on solutions to problems*".

In GLARE, a CIG can be represented as a hierarchical graph, where nodes are the actions to be executed, and arcs are the control relations linking them. GLARE distinguishes between *atomic* and *composite* actions (plans), where atomic actions represent simple steps in a CIG, and plans represent actions that can be defined in terms of their components via the has-part relation.

GLARE adopts five types of atomic actions:

- *work actions*, i.e. actions that describe a procedure which must be executed at a given point of the CIG,
- *pharmaceutical actions*, specifying a drug (or drug category) to be administered to the patient, and the dosage,
- *decision actions*, used to model the selection among different alternatives,
- *query actions*, i.e. requests of information (typically of patient's parameters),
- *conclusions*, which explicitly identify the output of a decision action.

In this paper, we focus on composite actions, and work and pharmaceutical atomic actions.

Actions in a CIG are connected through *control* relations. Control relations establish which actions can be executed next and in what order. In particular, the *sequence* relation explicitly establishes what the following action to be executed is; the *alternative* relation describes which alternative paths stem from a decision action, and the *repetition* relation states that an action has to be

repeated several times. The *constrained* relation is used in order to express more complex temporal relations between actions. In GLARE it is possible to express precise and imprecise dates, durations, delays, and complex forms of repetitions (Anselma et al. 2006). For the sake of simplicity, in this paper we adopt an easier approach for repetitions: we suppose that the exact number of repetitions of repeated actions is known, and explicitly express the constraints between repetitions using the above language.

2.2 Ontology of Interactions

In a recent work (Piovesan et al. 2014) we detailed our preliminary semantic model for the description of CIG actions and for the non-temporal interactions occurring between them. For the sake of brevity, in the left part of Figure 1we show a fragment of such an ontology relevant to this paper. In our ontological model, we focused on the goals of the actions and the drugs administered by the pharmaceutical actions, which are important sources of interactions between CIGs.

In the ontology, a work, pharmaceutical or composite action is described according to one or more relations *aimsTo* with its goals, called *intentions*, which are represented as *variations* of the patient status. Each variation relates to exactly one *attribute* (describing the patient status) and exactly to one *modality* (of the variation). For instance, the intention "*Decrease Blood Pressure*" is modelled by the variation of the attribute "*Blood Pressure*" with modality "*Decreasing*".

Intentions are organized along a hierarchy of ISA and PART-OF relations (not shown in Figure 1): high-level intentions can be broken up into lower-level intentions, and alternative decompositions are possible. For instance, the intention "*Decrease Blood Pressure*" can be decomposed into the alternatives "*Decrease Blood Volume*", "*Inhibition of Angiotensin Converting Enzyme (ACE)*", "*Block of Calcium Channels*" and so on.

In addition, pharmaceutical actions are described by the relation *substance* with the *drug* (or *drug category*) they recommend. Drug categories and drugs (the bottom level) are hierarchically organized and each level of the hierarchy is related(*has_effect*) to its effects, which are defined as *variations* of the patient status. For the drug taxonomy, the ATC

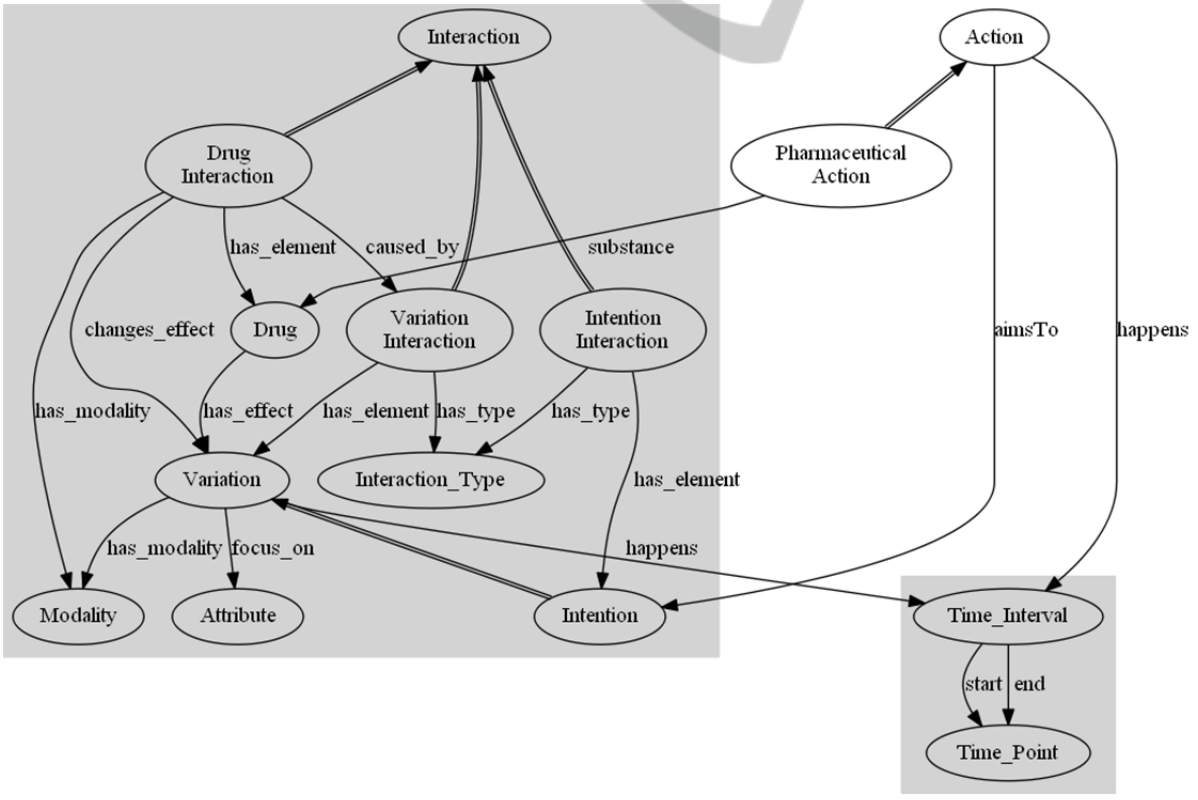

Figure 1: Preliminary semantic model. Double-line arcs represent is-a relations.

classification (http://www.whocc.no/atc/) is used; however, our approach is independent of the classification adopted. A distinguishing feature of our approach is that it copes with interactions at three different levels of abstraction: between the intentions of actions, between drug categories and between specific drugs (see concrete examples in (Piovesan et al. 2014)).

Intention interactions are described by the relation *has_element*, with the two intentions they involve and by an *interaction type*, whose basic values are *concordance*, *discordance* and *independence*. However, further refinements are possible, such as *opposite* for interactions focusing on the same attribute, but discording in the modality.

Drug interactions, besides the two drugs or drug categories they involve, are related to an effect (of one of the two drugs) and to the variation the interaction causes in such effect. Often, an interaction between two drugs is caused by an interaction between two of their effects. In order to model such information, the property *caused by* (optional) relates a drug interaction to a *variation interaction* (described by the two variations it involves and by a type). For instance, a drug interaction between the drugs nalidixic acid and calcium carbonate is caused by the variation interaction between "absorption of nalidixic acid" (of the first drug) and "urine alkalinization" (of the second), and its result is a decrease of the antibiotic absorption. Such example is detailed further in Sections 3 and 4.

It is worth stressing that drug interactions are independent of a specific guideline and of a specific action because they do not involve actions. When a new CIG is introduced in the knowledge base, introducing the specifications of all the interactions between its actions and the CIGs already stored is not needed. Only the relations between the actions of the new CIG and their intentions in the ontology and (in the case of pharmaceutical actions) the drugs they recommend must be pointed out.

Such ontological representation of intentions and effects allows the adoption of algorithms that, navigating the ontology, automatically infer the types of many interactions between intentions or drugs. More precisely, we implemented our ontology using OWL DL (http://www.w3.org/TR/owl2-overview/) and we expressed such a kind of basic medical knowledge about interaction recognition using Semantic Web Rules (http://www.w3.org/Submission/SWRL/). However, since not all the interactions can be inferred (especially for drug interactions) from the model, they can also be imported from external data sources.

Interactions may occur at all the levels of detail adopted in CIGs. At a high level of detail, usually actions are composite, thus intention interactions may occur. On the other hand, going towards lower levels of detail, pharmaceutical actions prescribe the administration of drugs (usually drug categories, from which the physician can choose, depending on the specific patient conditions) and, at this level, drug interaction occurs. Thus, in our opinion, a "black box" system pointing out all the possible interactions between two CIGs (considering all the possible levels of detail) would be not practically useful for physicians, since, in general, it would return too many interactions. In our previous work (Piovesan et al. 2014), we have devised a system that, collaborating with the physician to focus only on relevant parts of CIGs at the desired level of detail, helps her/him in the detection of relevant interactions (see Section 2.3), but we have neglected the temporal dimension. Modelling time, and extending the detection interaction system to cope with the temporal dimension, are the goal of this paper.

2.3 Non-temporal Interaction Detection

In the approach previously described, we have also proposed a flexible and interactive detection tool allowing physicians to navigate through the different abstraction levels. For instance, at the highest level, a physician may want to start to consider only the interactions between the intentions of the "top-level" actions of the guidelines. Then, focusing on a specific part of the guideline, (s)he may want to move down to a more detailed analysis, considering the decomposition of the composite actions into their parts, and/or the specific drugs category considered in order to reach the high-level intentions. In general, such approach provides physicians with the possibility of moving in both directions, i.e., focusing down from a general to a more specific analysis, or moving up, from a specific analysis to a higher level of abstraction. Additionally, the interaction detection algorithm maintains organized in a tree data structure (the navigation tree) the history of the focusing process, supporting both the addition of new CIG focuses, and the rollback to upper focuses. Each node of the tree consists of three main components: two pairs $<CIG_1, focus_1>$, $<CIG_2, focus_2>$ determining the desired level of abstraction and the focused actions, and an interaction component, in which, for each pair

$<A_i,A_j>$ of actions ($A_i \in focus_1$, $A_j \in focus_2$), the interactions between their intentions (or of the drugs they administer, in the case of pharmaceutical actions) are pointed out.

For the sake of brevity and simplicity, with no loss of generality, in the following we suppose that just two actions (one in the first CIG and one in the second CIG) are focused on by the user-physician, at the chosen level of detail.

3 TEMPORAL REPRESENTATION

3.1 Temporal Ontology

Coping with time in the interaction detection is of fundamental importance. Indeed, many of the entities involved in such a task are characterized by time, and physicians must consider such information when they execute more than one CIG.

In particular, actions are characterized by the time when they occur (or should occur), intentions are characterized by the time the physician expects they will be accomplished and effects (of drugs) are characterized by the time when they should happen. On the right side of Figure 1,we show how we relate such temporal information to the previous model. In particular, we introduce the relation *happens*, which relates an action or a variation to the *time interval* in which it takes place. A time interval is itself described by two *time points*, which represent the time when the interval starts and ends.

Obviously, the various times are strictly related to each other (i.e., the time of the effect of a drug depends on the time of administering such drug). In order to represent such relations, we detailed in our model two types of constraints: *qualitative* (such as, e.g., *before*, *after*, *during* (Allen 1983; Vilain et al. 1990)) and *quantitative* ones (such as, e.g., *duration*, *delay* and *date*). Notice that we support also imprecise quantitative constraints: for example, if the exact duration is not known, it is possible to express a minimum and a maximum duration (see Figure 2).

3.2 Temporal Constraint Representation

As discussed in the introduction, to deal with CIG interactions, three different sources of temporal constraints must be taken into account. In this section, we show how they can be represented in our model.

(1) Knowledge about (i) the delay (with respect to the action execution/drug administration) and (ii) the duration of effects (or intentions). In many cases, such data can be approximately predicted. In our model, they are represented with two quantitative constraints: (i) is a delay between the ending (or, in some cases, the starting) point of the action and the starting point of the effect (or intention); (ii) is a duration between the starting and ending point of the effect (or intention). In our approach, such knowledge is directly expressed at the ontological level.

Example 1. Calcium carbonate is a gastric antacid and it is often prescribed in order to alleviate the symptoms of gastroesophageal reflux after meals, when needed. One of its effects is the urine

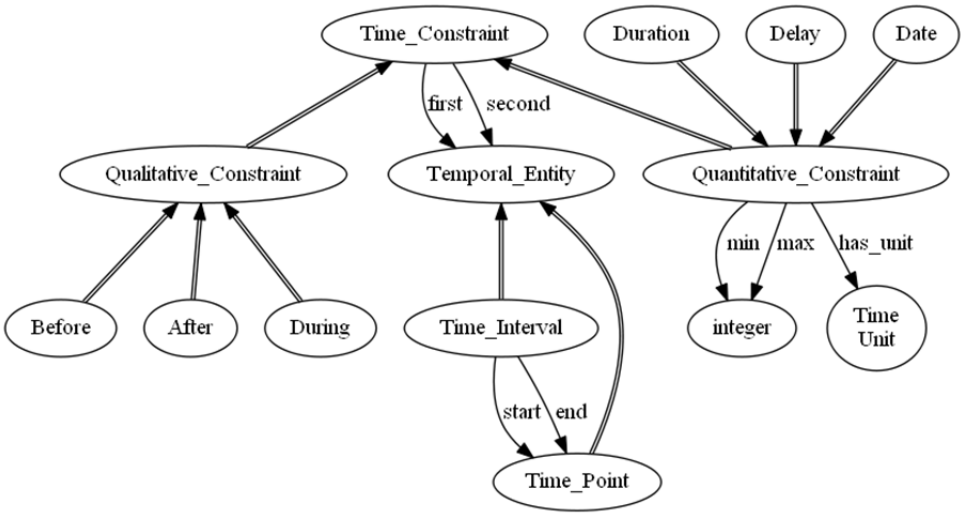

Figure 2: Temporal constraint ontology.

alkalinization (variation, modelled as an increase of urine pH), which starts at most one hour after the assumption and lasts 4-5 hours. In Figure 3, we show how we express temporal constraints between the calcium carbonate administration, which happens in a time interval (CATI) of which only the end (CAE) is relevant for the example, and the urine alkalinisation, which is characterized by a time interval (UATI) with a start (UAS) and an end (UAE) time points. For the sake of clarity, we do not represent the time unit of hours.

(2) Temporal constraints between actions in the CIGs: different types of constraints between CIG actions can be expressed in GLARE. All of them can be expressed through the model presented in this paper. In particular, the duration of an action can be expressed as a possibly imprecise quantitative constraint between its starting and ending points. Temporal constraints between two actions can be represented both through qualitative and quantitative constraints. Such constraints are directly represented in the CIG, as described in (Anselma et al. 2006). In particular, the constraint formalism has been designed in such a way that only the qualitative constraints that can be mapped to conjunctions of STP constraints (Dechter et al. 1991) can be expressed (see Section 4.2).

(3) Information about the time of execution of previous CIG actions on the specific patient: such information are modelled as absolute dates (expressed as distances from the start of the calendar). Also imprecise starting times and ending times are supported. Obviously, no execution time is provided in case an "abstract" (i.e., patient-independent) analysis of the interactions between CIGs must be performed.

Additionally, for the sake of generality, we also allow the possibility of expressing any constraint in the above language (constraints between CIG actions) also between action execution times.

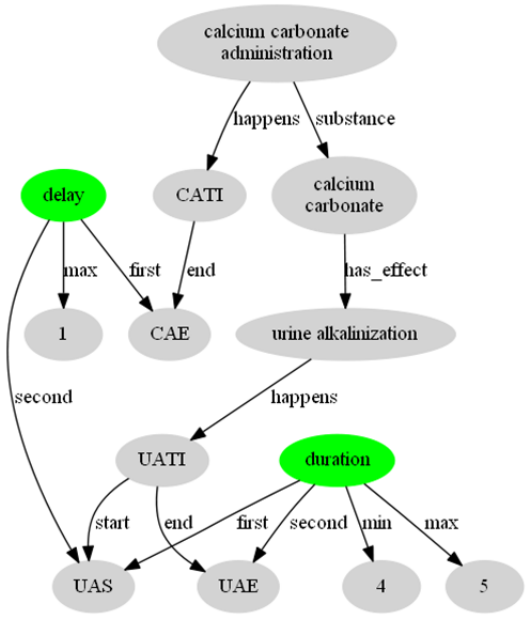

Figure 3: Representation of temporal constraints for the effect "urine alkalinization".

4 TEMPORAL REASONING

The goal of this work is to provide user-physicians with a general set of facilities in order to enable them to look for temporal interactions between CIGs. In this section, we introduce such facilities, the constraint propagations techniques we propose and how the facilities are grounded on the constraint propagation techniques.

4.1 Facilities

We provide the following facilities, where the user-physician is enabled to:

1. (*Interaction?*) check whether two actions in two different CIGs may interact, certainly interact or certainly do not interact;
2. (*Interaction (what-if)?*) assume a hypothetical execution time for some future actions and check whether – given such an assumption – two actions in two different CIGs may interact, certainly interact or certainly do not interact;
3. (*Time of future actions to have (or to avoid) an interaction?*) determine the execution times of some future actions in order to have or to avoid some interactions;
4. (*Time of future actions to have (or to avoid) an interaction (what-if)?*), as (3), but assuming some temporal constraints concerning the execution of future actions.

Notice that the answers may be not crisp, in the sense that an interaction between two actions can be temporally necessary, temporally possible or temporally impossible.

Taking into account the different contexts in which we support the temporal analysis of interactions (i.e., either considering the guidelines with no reference to specific patients or considering the actual execution of two CIGs on specific patients) and the specific temporal assumptions that we can have on the temporal data of the executions, we singled out three scenarios. In fact, different scenarios can induce different types of facilities available to

physicians. The first scenario is the "*no temporal log*" scenario, where no temporal information on the execution of the CIGs is available. This could happen because either the CIGs have not been executed yet or no time has been recorded. The second scenario is the "*temporally exact log*", where the times when the actions of the CIGs have been executed are known with the precision allowed by the granularity chosen for the log (e.g., hour or minute). In this scenario, for example, we assume that personnel records the exact time (e.g., hour or minute) of the start and of the end of the executed actions. The last scenario is the "*temporally imprecise log*", where, because of imprecision in time measurement or because of lack of information, the log does not contain the exact start/end time of the clinical actions but, e.g., a range of time when the action has started and a range of time when the action has ended.

In Table 1 we report the facilities available in each scenario, as detailed below.

4.2 Temporal Reasoning

Our treatment of temporal constraints is grounded on the STP framework (Dechter et al. 1991). In short, in STP a set of constraints is modelled as a conjunction of bounds on differences of the form $c \leq x - y \leq d$, which have an intuitive temporal interpretation, namely that the temporal *distance* between the *time points* x and y is between c (minimum distance) and d (maximum distance). Also strict inequalities are possible (i.e., <), and $-\infty$ and $+\infty$ can be used to denote infinite lower and upper bounds respectively.

Temporal reasoning on STP can be performed/computed by an "all-pairs shortest paths" algorithm such as the Floyd-Warshall's one. Such an algorithm provides as output the *minimal network* of the constraints, i.e., the minimum and maximum distance between each pair of points. A draft version of Floyd-Warshall's algorithm is shown below, where $1, \ldots, N$ denote the time points (e.g., starting/ending points of actions), *D[i,j]* represents the distance (difference) between *i* and *j*, and *Min* is the function which provides the minimum between the two arguments.

```
For k:=1 to N do
  For i:=1 to N do
    For j:=1 to N do
      D[i,j]=Min(D[i,j],D[i,k]+D[k,j])
```

Property. Floyd-Warshall's algorithm operates in a time cubic in the number of time points, and is correct and complete on STP (meaning that it performs all and only the correct inferences while propagating the STP constraints) (Dechter et al. 1991).

As mentioned above, we have chosen to design our ***high-level language for temporal information*** in such a way that all the temporal constraints can be mapped onto the STP framework. In particular, our temporal constraint language allows one to express both *quantitative constraints* such as (i) exact or imprecise (min/max) dates, (ii) exact or imprecise durations, (iii) exact or imprecise delays; and *qualitative constraints* between time points (e.g., P1 before P2) and/or time intervals (e.g., I1 during I2) (restricting the language to qualitative constraints mappable onto STP; see (Brusoni et al. 1997)). In our approach, such a high-level language is homogeneously adopted to represent (1) temporal constraints between actions in the CIGs; (2) exact dates of actions in the log, or temporal constraints between them; (3) temporal constraints in the ontology, and (4) temporal constraints on the hypothesized actions, if any.

The ***translation*** of the constraints of our high-level language into STP is easy: dates are mapped

Table 1: Facilities for temporal interaction detection and reasoning.

Query / ***Scenario***	*Interaction?*	*Interaction (what-if)?*	*Time of future actions to have (or to avoid) an interaction?*	*Time of future actions to have (or to avoid) an interaction (what-if)?*
No temporal log	N/A	HYP_TR(O, G1,G2,Var1, Var2,Hyp)	TR(O,G1,G2, Var1,Var2)	HYP_TR(O,G1,G2, Var1,Var2,Hyp)
Temporally exact log	TR(O,G1,G2, Var1,Var2,Log)	HYP_TR(O, G1,G2,Log, Var1,Var2,Hyp)	TR(O,G1,G2, Var1,Var2,Log)	HYP_TR(O,G1,G2, Var1,Var2,Log,Hyp)
Temporally imprecise log	TR(O,G1,G2, Var1,Var2,Log)	?	TR(O,G1,G2, Var1,Var2,Log)	?

into distances with respect to a fixed Reference Time (e.g., the start time of the calendar), durations into distances between ending and starting points, delays into distances between points. Also the translation of qualitative constraints is easy: just as an example, *I1 during I2* corresponds to the set of STP constraints $\{0 < Start(I1) - Start(I2) \leq +\infty, 0 < End(I2) - End(I1) \leq +\infty\}$.

Property. The translation of each constraint in our high-level temporal language into STP can be performed in constant time.

In order to provide the temporal facilities in Table 1, the first step is the ***collection of*** (relevant) ***constraints*** from the log (if present), from the CIGs and from the ontology. In the case of exact temporal log, each executed action is timestamped with its starting and ending time (which are exact dates); in the case of temporally imprecise log, the log explicitly contains temporal constraints between the executed actions. In both cases, temporal constraints are simply collected by inspecting the log. The collection of constraints from the CIGs involves the navigation of the CIGs (expressed in GLARE) from the starting action A_s to the focused action A_f, and the collection of the constraints on the arc in the path connecting them. In case multiple alternative paths are present, each one of the paths must be considered independently of the others (in the rest of the discussion, for the sake of simplicity, we assume that only one path is considered). Additionally, in case composite actions are present in the path, also the constraint that the temporal extent of a composite action contains the extents of its components must be considered. Finally, the ontology can be easily navigated in order to retrieve the temporal constraints between the focused actions and their focused effects. Different types of arcs in the ontology have to be navigated, depending on the types of the focused action. Figure 3 shows the case of pharmacological prescriptions (*calcium carbonate administration* in the example). The *happens* and *end* (or *start*) arcs connect actions with the ending (starting) point of the time when they occur (e.g., *CAE* in Figure 3). The *substance* arc connects pharmacological actions to the drug they prescribe, and the *has_effect* arc points out the effect (variation) caused by such a drug. In turn, *happens* and *start*/*end* arcs relate effects to their starting/ending times (e.g., *UAS*, *UAE* in Figure 3). Temporal constraints between such endpoints (e.g., the *delay* between *CAE* and *UAS*) can then finally be retrieved.

After the collection of constraints (from log, CIGs and ontology) is performed, and all constraints are translated onto STP constraints, temporal reasoning can be performed, to offer the above facilities to user-physicians.

To provide the different facilities shown in Table 1 we rely on two basic algorithms that propagate the temporal constraints: *TR*, which performs temporal reasoning, i.e., it checks for consistency and evaluates the minimal network using Floyd Warshall's algorithm, and *HYP_TR*, which performs temporal reasoning assuming some hypothetical temporal information. The parameters *O, G1, G2, Var1, Var2, Log, Hyp* in the table represent the ontology, the two CIGs, the two interacting variations to be examined, the log and the hypothetical temporal constraints, respectively.

Now we discuss how the different types of log (log with no temporal information, with exact times, and with imprecise temporal information) affect the facilities.

When no information is available on the execution of the CIGs ("no log", first row of the table), all relative temporal relations between the two CIGs are possible. Therefore, in order to infer any meaningful conclusion on the interactions, it is necessary to *anchor* a CIG to the other, otherwise the query cannot be answered (N/A in the table). Such anchoring can be made in two ways in the "no log" scenario: by devising an interaction between the two CIGs (in *Time of future actions to have (or to avoid) an interaction?*) or by assuming some temporal relations between the two CIGs in the facilities that contemplate hypothetical temporal constraints.

When precise temporal information is available on the execution of the CIGs ("temporally precise log", second row of the table), all types of queries can be answered. Since we know the exact time when the actions have been performed, it is possible to check whether they interact in time. Notice that temporal reasoning is required also in this case: in fact, the time of "future" actions, i.e., the time of actions in the CIGs not yet performed, is not exactly known. Therefore, the temporal constraints in the CIGs, along with the temporal constraints from the logs, have to be propagated.

When temporal information on the execution of the CIGs is available but it is imprecise ("temporally imprecise log", third row of the table), it is important noting that hypothetical queries may have some undesired side effect. In fact, in hypothetical queries, where some hypothetical temporal constraints are added to the known temporal information, the propagation of such new temporal information could cause a tightening of some imprecise log constraints.

In this case, such constraints could take only some of the possible values that make the hypothetical query consistent. However, these constraints are not "controllable", in the sense that they represent imprecision in the measurement of the time and it is not possible for the user to choose a specific time value. Treating this case is an open problem and it is left as a future work.

For the sake of brevity, we illustrate in more detail only the facility *Hypothetical Interaction?* in the "temporally exact log" scenario (see Algorithm 1). After extracting the temporal constraints from the CIGs, from the logs and from the ontology in a STP, the hypothetical temporal constraints are provisionally added to the STP. Then the constraints are propagated and the resulting minimal network is used to answer the query. Such minimal network, in fact, contains the strictest inferred constraints between the two variations under consideration. Thus, by examining the inferred temporal difference between the starting and ending points of variations *Var1* and *Var2*, we can determine whether their overlap is certain, possible or it is certain that there is no overlap.

As regards the evaluation of the algorithm, its computational complexity is dominated by HYP_TR, which operates in time cubic in the number of (i) the actions considered in the two CIGs plus (ii) the actions in the log plus (iii) the hypothesized actions.

Example 2. We consider the case where a patient suffering from gastroesophageal reflux treated with calcium carbonate (see Example 1) contracts a urinary tract infection and, thus, the two pertaining CIGs have to be executed at the same time on this patient. In particular, the urinary tract infection is treated with nalidixic acid, which starts its "absorption of nalidixic acid" effect (modelled as an increase of nalidixic acid blood level) in at most one hour after the assumption and lasts 4 hours. We consider the case where the physician wants to know if the administration of nalidixic acid will interact with the assumption of calcium carbonate. The physician decides to focus in the CIGs on the pharmaceutical actions of administration of the two drugs. We assume that the patient takes the calcium carbonate after each meal (say lunch at 1 pm and dinner at 8 pm). The physician decides to perform a "what-if" analysis of interaction and to explore the consequences of administering the nalidixic acid at 3 pm and (s)he asks to the system if the two focused actions interact. First, a non-temporal interaction is extracted from the ontology between the two drugs, *caused by* a variation interaction between the "urine alkalinization" and the "absorption of nalidixic acid" effects, with *has_modality* "decreasing" of the "absorption of nalidixic acid" effect. Then, in order to decide if the two actions temporally interact, the facility *Hypothetical Interaction* is used, with parameters the ontology, the two CIGs, the two interacting variations "urine alkalinization" and "absorption of nalidixic acid", and the hypothesis of administration of nalidixic acid at 3 pm. The propagation of the temporal constraints allows the physician to discover that the calcium carbonate has effect surely between 2 pm and 5 pm and that the temporal intervals of effect of the two interacting drugs surely overlap at least from 4 pm to 5 pm. Thus, the facility returns YES. Because of this result, in order to avoid the interaction, the physician can decide to change one of the two drugs or the time of administration of the antibiotic, repeating the focusing and detection process.

```
Hypothetical Interaction?(O, G1, G2,
Var1, Var2, Hyp)
  Extract temporal constraints
  HYP_TR on temporal constraints given
    hypothesis Hyp
  Given minimal network:
  If there is necessarily an overlap
    between variation Var1 and variation
    Var2 then return YES
  Else If variation Var1 necessarily
    does not temporally overlap
    variation Var2 then return NO
   Else return MAYBE
```

Algorithm 1: Algorithm for detecting temporal interaction assuming some temporal constraints.

5 RELATED WORKS AND CONCLUSIONS

The treatment of comorbid patients is one of the main challenges for the modern healthcare. This is a hot topic in Medical Informatics, too, and several approaches are recently emerging.

The approach in (Michalowski et al. 2013) and (Wilk et al. 2013) uses constraint logic programming to identify and address adverse interactions. In this solution, a constraint logic programming (CLP) model is derived from the combination of logical models that represent each CIG, then a mitigation algorithm is applied to detect and mitigate interactions. On the other hand, Sánchez-Garzón et al. (Sánchez-Garzón et al. 2013) propose an agent-based approach to guideline merging. Each guideline

is considered as a physician expert in the treatment of a single disease, and is represented by an agent with hierarchical planning capabilities. The result is obtained through the coordination of all the agents, and respects the recommendations of each guideline.

Riaño et al. represent guidelines as sets of clinical actions that are modelled into an ontology (López-Vallverdú et al. 2013). To combine two treatments, first they are unified in a unique treatment and then a set of "combination rules" is applied to detect and avoid possible interactions. A model-based automatic merge of CIGs is then purposed in (Riaño and Collado 2013), through the definition of a combining operator. Jafarpour and Abidi (Jafarpour and Abidi 2013) use semantic-web rules and an ontology for the merging criteria. Given these, an Execution Engine dynamically merges several CIGs according to merge criteria. GLINDA proposes a wide ontology of cross-guideline interactions (http://glinda-project.stanford.edu/guidelineinteractionontology.html). We recently proposed an original approach, supporting user-driven and interactive interaction detection over different levels of abstractions (Piovesan et al. 2014).

However, although interactions can only occur in time, to the best of our knowledge no previous approach to the treatment of interactions (and comorbidities) has already provided facilities to address the temporal dimension. This is the goal of the approach in this paper, in which we proposed a general approach, suitable in different situations (e.g., either in case a specific comorbid patient is considered, or in case "abstract" possible interactions between CIGs are taken into account), and providing a wide range of facilities to user-physicians.

Temporal issues are pervasive in the CIG context and many previous approaches have faced some of them (see, e.g., the survey in (Terenziani et al. 2008)). In particular, in the Asbru (Shahar et al. 1998) and in the GLARE (Anselma et al. 2006) projects, rich representation formalisms have been proposed to cope with temporal constraints in the CIGs, and in GLARE correct and complete temporal constraint propagation algorithms have been proposed to reason with them and to merge them with the time of execution of actions on specific patients (Anselma et al. 2006). However, to the best of our knowledge, no other approach to CIGs has explicitly addressed the treatment of time and temporal constraints for the detection of CIG interactions. In this sense, we believe that our approach, besides being innovative, is somehow *complementary* with respect to several other approaches in the literature, so that an integration with them can be devised as a future work (e.g., with Riaño's methodology to merge CIGs (Riaño and Collado 2013)).

We are currently developing a prototypical implementation to demonstrate our approach, based on GLARE. In our short-term future work, we aim at extending our approach to cope also with cases not covered in Table 1. In our long-term future work, we will attempt to support physicians also in the interaction solving, and, finally, in merging multiple guidelines in the treatment of a specific patient.

ACKNOWLEDGEMENTS

The work described in paper was partially supported by Compagnia di San Paolo, in the Ginseng project.

REFERENCES

Allen, J.F., 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11), pp.832–843.

Anselma, L. et al., 2006. Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines. *Artificial Intelligence in Medicine*, 38(2), pp.171–195.

Brusoni, V. et al., 1997. Later: managing temporal information efficiently. *IEEE Expert*, 12(4), pp.56–64.

Dechter, R., Meiri, I. and Pearl, J., 1991. Temporal Constraint Networks. *Artif. Intell.*, 49(1-3), pp.61–95.

Fridsma, D.B., 2001. Special Issue on Workflow Management and Clinical Guidelines. *Journal of the American Medical Informatics Association*, 22(1).

Gordon, C. and Christensen, J.P. eds., 1995. *Health telematics for clinical guidelines and protocols*, Amsterdam, Netherlands: IOS Press.

Horvitz, E., 1999. Uncertainty, Action, and Interaction: In Pursuit of Mixed-Initiative Computing. *IEEE Intelligent Systems*, 14(5), pp.17–20.

Jafarpour, B. and Abidi, S.S.R., 2013. Merging Disease-Specific Clinical Guidelines to Handle Comorbidities in a Clinical Decision Support Setting. In *AIME*. pp. 28–32.

López-Vallverdú, J.A., Riaño, D. and Collado, A., 2013. Rule-Based Combination of Comorbid Treatments for Chronic Diseases Applied to Hypertension, Diabetes Mellitus and Heart Failure. In R. Lenz et al., eds. *Process Support and Knowledge Representation in Health Care*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 30–41.

Michalowski, M. et al., 2013. Using Constraint Logic Programming to Implement Iterative Actions and Numerical Measures during Mitigation of Concurrently Applied Clinical Practice Guidelines. In

N. Peek, R. M. Morales, and M. Peleg, eds. *Artificial Intelligence in Medicine*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 17–22.

Peleg, M., 2013. Computer-interpretable clinical guidelines: A methodological review. *Journal of Biomedical Informatics*, 46(4), pp.744–763.

Piovesan, L., Molino, G. and Terenziani, P., 2014. An ontological knowledge and multiple abstraction level decision support system in healthcare. *Decision Analytics*, 1(1).

Riaño, D. and Collado, A., 2013. Model-Based Combination of Treatments for the Management of Chronic Comorbid Patients. In *Artificial Intelligence in Medicine*. 14th Conference on Artificial Intelligence in Medicine. Springer, pp. 11–16.

Sánchez-Garzón, I. et al., 2013. A Multi-agent Planning Approach for the Generation of Personalized Treatment Plans of Comorbid Patients. In *AIME*. pp. 23–27.

Shahar, Y., Miksch, S. and 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), pp.29–51.

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

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

Vilain, M., Kautz, H. and van Beek, P., 1990. Constraint propagation algorithms for temporal reasoning: a revised report. In D. S. Weld and J. de Kleer, eds. *Readings in Qualitative Reasoning About Physical Systems*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 373–381.

Wilk, S. et al., 2013. Mitigation of adverse interactions in pairs of clinical practice guidelines using constraint logic programming. *Journal of biomedical informatics*, 46(2), pp.341–353.