Temporal Conformance Analysis and Explanation on Comorbid Patients

Luca Piovesan, Paolo Terenziani and Daniele Theseider Dupré
DISIT, Computer Science Institute, Università del Piemonte Orientale, Italy

Keywords: Computer-interpretable Clinical Guidelines, Comorbidities, Conformance Analysis, Answer Set Programming.

Abstract: The treatment of comorbid patients is one of the main challenges of modern health care, and many Medical Informatics approaches have been devoted to it in the last years. In this paper, we propose the first approach in the literature that analyses the conformance of execution traces with multiple Computer-Interpretable Guidelines (CIGs), as needed in the treatment of comorbid patients. This is a fundamental task, to support physicians in an a-posteriori analysis of the treatments that have been provided. Notably, the conformance problem is very complex in this context, since CIGs may have negative interactions, so that in specific circumstances full conformance to individual CIGs may be dangerous for patients. We thus complement our conformance analysis with an explanation approach, aimed at justifying deviations in case they can be explained in terms of interaction management, e.g., some possible undesired interaction has been avoided. Our approach is based on Answer Set Programming, and, to face realistic problems, devotes specific attention to the temporal dimension.

1 INTRODUCTION

Clinical practice guidelines are one of the major tools that has been introduced to grant both the quality and the standardization of healthcare services, on the basis of evidence-based recommendations. The adoption of computerized approaches to acquire, represent, execute and reason with Computer-Interpretable Guidelines (CIGs henceforth) can provide crucial additional advantages. Therefore, in the last twenty years, many different approaches and projects have been developed to manage CIGs (consider, e.g., the book (Ten Teije et al., 2008) and the survey (Peleg, 2013)). By definition, clinical guidelines address specific clinical circumstances (i.e., specific pathologies). However, individual patients may be affected by more than one pathology (comorbid patients). The treatment of such patients is one of the main challenges for modern health care, also due to the aging of population, and the increase of chronic pathologies. In fact, in comorbid patients the treatments of single pathologies may interact with each other, and the approach of proposing an ad-hoc "combined" treatment to cope with each possible comorbidity does not scale up: *"Developing Clinical Practice Guidelines that explicitly address all potential comorbid diseases is not only difficult, but also impractical, and there is a need for formal methods that would allow combining several disease-specific clinical practice guidelines in order to customize them to a patient"* (Michalowski et al., 2013). Thus, new methodologies are required to study the interactions between treatments, and to combine treatments: *"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). In the last years, several computer-based approaches have started to face this problem, aiming at providing physicians with different forms of support for managing multiple CIGs and their interactions.

As part of the previous work in this area, we devised several methodologies to support physicians in the management of comorbid patients: physician-driven navigation of CIGs at different levels of abstraction, to focus on the parts that are relevant for potential interactions (Piovesan et al., 2015); knowledge-based detection of interactions between actions (Anselma et al., 2017); mixed-initiative management of detected interactions (Piovesan and Terenziani, 2015); merging the CIGs into a unique treatment for a given patient (Piovesan and Terenziani, 2016).

In this paper, we ground on the above mentioned previous work and on an approach to check *a-posteriori* conformance of the treatment of a patient with respect to *one* clinical guideline (Spiotta et al.,

Piovesan, L., Terenziani, P. and Dupré, D.
Temporal Conformance Analysis and Explanation on Comorbid Patients.
DOI: 10.5220/0006535400170026
In *Proceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2018) - Volume 5: HEALTHINF*, pages 17-26
ISBN: 978-989-758-281-3

2015; Spiotta et al., 2017), to face the comorbidity problem from a new perspective, that, to the best of our knowledge, has not been considered yet. We explore the interplay between CIGs from the viewpoint of *a posteriori conformance analysis* (Groot et al., 2009), intended as the adherence of an observed CG execution trace to the CIGs executed on a (comorbid) patient. Our goal is not to provide an evaluation of whether the treatment was appropriate or not; rather, we identify actions that have been executed or could have been executed, following the individual CIGs, and may interact, according to the approach in (Anselma et al., 2017). We then allow for interpreting the actual trace according to possible ways for managing interactions (defined in (Piovesan and Terenziani, 2015)). Therefore, the trace may deviate from a strict application of the individual CIGs under execution. Notably, we also identify and point out cases where the execution log is conformant with all the CIGs, but some interaction has not been avoided: in such cases, the adherence to the CIGs might not have been the best option. Reasoning on interactions in a-posteriori conformance analysis may complement reasoning on the same knowledge to *support* CIG execution: for example, it may be used (Quaglini, 2008) at discharge time to support documenting the patient care process, while, at execution time, physicians may not have time to interact with a support tool.

Significant conclusions can be drawn from this conformance analysis if the trace is reasonably correct and complete as regards patient data and executed actions. Missing events in a trace, or missing data in the description of an event (or even the incorrect recording of events), can indeed be hypothesized to have occurred or to hold, in order to make conformance analysis more flexible (Chesani et al., 2016); however, a high number of incorrect or missing events in the trace with respect to actual events, in combination with several ways (studied in (Spiotta et al., 2015; Spiotta et al., 2017) and in the present paper) for explaining discrepancies between CIG and trace, would make the space of possible explanations quite large. In this paper we concentrate on explaining discrepancies in terms of management of possible interactions; discrepancies that cannot be explained in this way are considered cases of non-conformance, which may be due to incompleteness or incorrectness of the trace, even though, in general, there is no way for distinguishing. On the other hand, it is not realistic to assume that at execution time explicit information is recorded on the fact that the actions that have actually been executed deviate from the recommendations of an individual CIG: detecting this is part of conformance analysis.

To make our approach more precise in the analysis of non-conformances due to possible interactions, we take into account the temporal dimension. Indeed, patient data hold at specific (intervals of) time; the timing of actions should respect temporal constraints in the CIGs, and interactions occur (or do not occur) in time. This makes conformance analysis more complex but richer: for instance, the CIG constraints may be violated in order to temporally avoid undesired interactions. Temporal analysis requires that at least imprecise temporal information on actions is provided; imprecision could lead an interaction, and then an explanation, to be considered possible (due to potential overlap of effects in time) in cases where more precise temporal information would not support that possibility.

In this paper, we propose a general methodology to cope with the above issues, that is mapped to Answer Set Programming (ASP), which, as shown in (Spiotta et al., 2015; Spiotta et al., 2017) for the case of a single CIG, is quite useful to analyse conformance, since, on the one hand, it supports the non-monotonic forms of reasoning naturally used by physicians in this context and, on the other hand, it naturally supports the search for alternative explanations.

In the following, we first describe (Section 2) the state of the art in AI support for comorbidities management and in conformance verification for CIGs. Then, in Section 3, we present more in detail the data/knowledge sources used in our analysis. Sections 4 and 5 are the core of the paper: we first introduce (Section 4) our general methodology to perform conformance analysis for concurrently executed CIGs, then (Section 5) we describe in detail the modalities for interaction management (taken from medical literature) that are considered in our approach to explain deviations from a strict application of individual CIGs. In Section 6 we demonstrate the potential of our approach on a relatively simple - but explicative - example. Finally, in Section 7, we summarize the contributions of our approach and point out possible future work.

2 BACKGROUND

Until now, the two main tasks that we homogeneously deal with in our approach have been only pursued in isolation by the approaches in the literature: (1) there are approaches to conformance analysis for CIGs, which only consider one CIG; (2) there are approaches to manage multiple CIGs and their interactions, to cope with comorbid patients, but none of such approaches face the conformance problem. In the following, we separately consider the state of the art on the two issues, only focusing on the work that is more closely related to ours.

In the last years, several AI approaches have been developed for supporting the treatment of **comorbid** patients (see the survey (Fraccaro et al., 2015)). For the detection of interactions between CIGs, the most closely related approach is the one in (Zamborlini et al., 2014; Zamborlini et al., 2017). It provides a CIG-independent conceptual model for medical actions and reasoning forms operating on it. Moreover, in such a work general rules are proposed in order to identify different types of interactions on the basis of such a knowledge.

Several approaches have been devoted to the generation of integrated CIGs. Some of them are not "conservative": adopting different techniques, they use the input CIGs as a starting point to build a mostly new CIG which has no undesired interactions; this is the case for the work in (Sánchez-Garzón et al., 2013), which uses an agent-based approach and hierarchical planning. Other approaches, including the one in GLARE (Piovesan and Terenziani, 2016), adopt more conservative techniques: since CIGs are evidence-based, physicians rely on them, so that interactions are managed with the minimum possible deviations from the original CIGs. Within such approaches, the one in (Wilk et al., 2013), uses constraint logic programming to identify and address adverse interactions, while (Wilk et al., 2017) is based on first-order logic and extends the above work for dealing with more than two CIGs; (Riaño and Collado, 2013) proposes a model-based approach for the combination of CIGs; (Jafarpour and Abidi, 2013) a semantic-web framework and (Zhang and Zhang, 2014; Merhej et al., 2016) ASP-based approaches.

A limited number of approaches have dealt with verifying conformance of a trace of actions with recommendations in a CIG. In (Groot et al., 2009), differences between actual actions and CG prescriptions are detected and analyzed, e.g., by comparing, for non-compliant actions, actual findings with findings that support the action according to the CIG. (Bottrighi et al., 2011) focuses on the interaction between clinical guidelines (CGs) and the basic medical knowledge (BMK) in the light of the conformance problem. The work in (Spiotta et al., 2015; Spiotta et al., 2017) also focused on the interaction between BMK and a CIG, using ASP, aiming at providing a justification for non-conformances.

3 PRELIMINARIES

At least four different types of data/knowledge sources should be considered to analyze compliance: patient data, traces of execution, CIG models, and general knowledge about action effects, intentions and interactions between such elements (henceforth, called *ontological knowledge*).

By **patient data** we mean *patients' findings*, i.e., data which are usually collected in patients' electronic health records (EHR). In particular, as discussed in the introduction, we intend that available data represent all known information that is relevant for treating the patient, and that such pieces of information are temporally tagged (possibly with imprecision). Also, the available **execution trace** (*trace* for short) is considered as all that is known on the clinical actions that have been executed on the patient. The occurrence of actions is temporally tagged with its start and end time. We also assume that, in the case of decisions, the log explicitly indicates the alternative that has been chosen.

As concerns the **CIG model**, our approach is not biased towards any specific CIG formalism; however, we will use the GLARE formalism (Terenziani et al., 2001) as a concrete example, due to its specific attention to the temporal aspects. Indeed, we simply consider the possibility of distinguishing between atomic and composite actions, and of specifying (therapeutic and diagnostic) decisions. CIGs specify the control flow of actions and include temporal constraints between them. Additionally, actions may have preconditions, and temporal constraints between the time when preconditions hold and the time when the related action must be executed can be specified. Actions are considered for execution as follows:

- When the control flow indicates that an action *a* will have to be executed (in a time window dependent on the temporal constraints in the CIG), we say that *a* becomes *scheduled*. This means that, in case we are considering a sequence of actions followed by a decision, all the actions in the sequence (and the decision) are scheduled, while t he actions following the decision are not, since they belong to alternative paths, and, at this point of the analysis, the physician could not know a-priori which alternative she will take.
- When an action is reached by the control flow, i.e., the previous action ends, it becomes *candidate* and its execution proceeds according with the execution model described in (Spiotta et al., 2015; Spiotta et al., 2017): a *candidate* action could become *active* or *discarded*; if *active*, it could either be *completed* or *aborted*. Here, we just remark

that an action should start (become *active*) at a time such that all preconditions, with their temporal constraints, enable the action, if such a time exists.

Ontological Knowledge. Possible interactions between CIGs, and between CIG recommendations and the patient status, have to be identified. To do so, we have to explicitly consider also the *effects* and *intentions* of actions, and the time window in which such elements can occur. Additionally, we must rely on a knowledge base that models the possible interactions between action effects and intentions, which is CIG and patient independent. To address such a need, we take advantage of the temporal *ontology* provided in (Anselma et al., 2017) to devise a decision support system helping physicians in the treatment of comorbid patients.

4 A GENERAL APPROACH TO CONFORMANCE ANALYSIS AND EXPLANATION

A high-level view of our general methodology is graphically shown in 1. For the sake of simplicity, we assume the execution of two CIGs and binary interactions (i.e., between pairs of actions), which is the case explicitly considered in (Piovesan and Terenziani, 2015). Our approach can be easily generalized to multiple CIGs as long as we only consider at each time the interaction of two of them, while (Piovesan and Terenziani, 2015), and then the approach in this paper, could be generalized to non-binary interactions.

We perform conformance analysis for the times when some change occurred, either in the state of actions (e.g., an action becomes candidate, or a candidate action becomes active) or in the state of the patient (e.g., a new value for a finding is detected). At each such time (indicated as *Reference Time* - RT - henceforth), we consider as input of our analysis:

1. The set of all actions that are scheduled, candidate or active, each one with its "execution window" (i.e., the range of time within which the action must be started and/or completed, given the constraints in the respective CIG);
2. The set of all the possible effects of such actions, each one with its "existence window" (i.e., the range of time in which the effect may start and end, given knowledge in the ontology of (Anselma et al., 2017));
3. The set of past actions with effects whose "existence window" includes RT;
4. The status of the patient at RT;
5. The trace of execution.

The sets (1) and (3) are the *relevant actions* for RT. Considering (1), (2), and (3), and ontological knowledge modeling possible interactions (Anselma et al., 2017) we detect whether interactions are possible between an action in (1) from one CIG and an action in (1) or (3) from the other CIG; i.e., at least one action should not have started or not be completed, so that some modification can be applied in order to manage interactions. In practice, for a given pair of actions, it is enough to perform the detection at the earliest RT for which they are both relevant. The interaction detection follows the general methodology devised in (Anselma et al., 2017) and takes into account the direct and indirect effects of the actions and whether some of such effects interact and may overlap in time. We consider two cases:

- No interaction is possible; in such a case, no deviation from the CIGs can be justified by the analysis of interactions. We check, slightly extending the methodology in (Spiotta et al., 2015; Spiotta et al., 2017), which copes with a single CIG, whether the (proper part of the) execution trace is conformant with the CIGs, and report, as non-justified, any non-conformance.
- One interaction is possible (the case of multiple interactions is significantly more complex, and we plan to address it as future work). In such a case, deviations from the CIGs might be explained as a way to manage the interaction (e.g., to avoid it). In the previous work in this area, (Piovesan and Terenziani, 2015) identified different *modalities* used by physician to manage interactions. This issue is described in more detail the next section.

5 EXPLAINING NON-CONFORMANCE

Physicians adopt different methodologies to manage interactions (e.g., avoid undesired interactions) between (the effects of) CIG actions. In (Piovesan and Terenziani, 2015) a set of "modalities" to achieve such a goal have been identified. Notably, such options are not mutually exclusive: indeed, in several practical cases, many options are possible, and the physicians have to choose between them. In the following, we describe how we check whether one of such modalities may be used to explain a non-conformance to the original CIGs aimed at managing a possible interaction. The first modalities aim at avoiding an interaction.

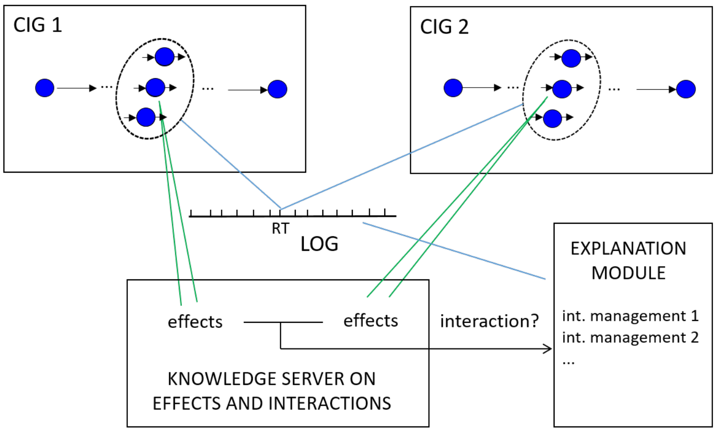

Figure 1: The possible interaction between two actions relevant for a reference time RT may be used to explain the rest of the log.

Replanning. One of the interacting actions is substituted by a new plan (set of actions), achieving the same goal, or a similar one, according to (Piovesan and Terenziani, 2015), but avoiding the interaction. Such an option explains cases of non-conformance in which an action in a CIG is not executed (while it should have been, given the conditions and constraints in the CIG), and one or more actions, not present in the CIGs, are executed (they are present in the trace). In order to justify such a non-conformance with the application of the replanning modality, our approach first detects whether an interaction would have occurred (following the CIG), and then uses ontological knowledge to check whether the actions in the trace but not in the CIGs reach the same goals (intentions) of the "substituted" action.
Temporal Avoidance. Interactions can be temporally avoided. In order to do so, interacting actions can be executed at times such that the interaction cannot actually occur (i.e., their effects do not overlap in time). Such a modality can be used in order to explain cases of non-conformance due to the violation of some of the temporal constraints in one or more CIGs. To do it, our approach checks whether, in case the CIG constraints had been respected, an interaction would have possibly occurred, while such interaction was not possible given the execution times in the trace.

Medical knowledge indicates that not all undesired interactions strictly need to be avoided. In some cases, CIGs can be adjusted to manage the situations in which the interactions arise. We support three main management options to this purpose: dosage adjustment (for drug interactions), effect monitoring, and interaction mitigation.
Dosage Adjustment. Interactions can be mitigated through a variation of dosage with respect to the ones recommended in the CIGs. Such a pattern can be easily identified (by comparing the dosage in the trace with the one in the CIG) and explained (by identifying the potential interaction, and using the ontology to check whether the sign of the dosage variation is the proper one for mitigating the interaction).
Effect Monitoring. In some cases, monitoring the effects of the interaction is enough. In particular, if an interaction causes a change of some parameters of the patient, they have to be monitored and evaluated by the physicians during the span of time in which the interaction occurs. Obviously, if a serious risk is detected, other management options can be finally applied. The effect monitoring modality explains traces in which (i) interacting actions present in the CIGs are indeed executed, but (ii) they are followed by a monitoring action (not present in the original CIGs) and a decision action (not present in the original CIG) to evaluate the state of the patient and decide whether to continue the current therapy or not. In the latter case, the trace must contain another management or the CIG must be suspended.
Interaction Mitigation. Some interactions cause undesired but tolerable side effects. In such cases, a new action (or set of actions) that mitigates such effects can be added to the interacting CIGs. Of course, the new action is a deviation with respect to the two CIGs. To explain such a deviation as an application of the interaction mitigation modality, the additional action must mitigate the effects of an occurring interaction.

Not all interactions between CIGs are negative or undesired. This is the case when two actions in the two CIGs pursue the same or similar goals. In such a case, intention alignment can be applied by physicians.
Intention Alignment. In the case of intention alignment, the physician may want to "merge" two actions of two different CIGs into a single one, executing it at a time which respects the temporal constraints of both CIGs, or to substitute them with a new action, which

pursues the same (or similar) goals of the two actions. This modality can be used to explain the occurrence in the trace of an action which is not present in any of the two CIGs, instead of two CIG actions. The ontological knowledge is used to check whether the new action can achieve both the goals of the actions it substitutes.

Besides the above modalities, other modalities have been identified in (Piovesan and Terenziani, 2015). Such modalities are practically useful, but detecting them is less useful in an a-posteriori analysis.

Safe alternative and **interaction alignment**. Such modalities consist in the avoidance (*safe alternative*) or enforcement (*interaction alignment*) of an interaction through the choice of alternative paths in the CIGs, when alternative therapeutic actions or paths of actions are specified. Given the trace, we can just recognize the paths chosen by the physicians. In principle, it could be hypothesized that, at the time of some decision, other paths have been disregarded to avoid or enforce interactions, but this is not useful to explain any deviation from the CIGs (since, indeed, paths in the CIGs have been carried on). Notably, in our current approach, we do not even try to detect whether some interaction (which has motivated some deviation from the CIGs) could have been avoided through the application of the safe alternative option (i.e., by selecting, a-priori, different paths from the CIGs). Such an analysis would be quite complex, and scarcely useful in an a-posteriori conformance analysis, because, in the clinical practice, it is not realistic to expect that physicians consider all the possible future consequences of their therapeutic choices, exploring in the CIGs all the paths stemming from each decision, and analysing all possible interactions between them; notably, such an analysis, though complex, may be very useful in the context of decision support.

We briefly explain in the following how the analysis is performed in ASP. A choice rule:

```
1 {management(Cg1,A,Cg2,B, no_management);
management(Cg1,A,Cg2,B, replanning);
management(Cg1,A,Cg2,B, temporal_avoidance);
... } 1 :-
possiblyInteractScheduled(Cg1,A,Cg2,B,_,_,S),
relevant(Cg1,A,S), relevant(Cg2,B,S),
{ended(A,S1) : S1<=S; ended(B,S2) : S2<=S} 1.
```

where different management modalities are considered in the conclusion, allows the ASP solver to consider a candidate answer set for each such modality. In general, the rule applies at a reference time S if: the actions are scheduled, they are not both completed, and they possibly interact, given the state of the execution at S and the temporal constraints in the CIGs. This is verified with the predicate `possiblyInteractScheduled`, which is defined based on the knowledge about effects and actions exported by the knowledge server in 1, and temporal reasoning implemented in ASP. From the knowledge server we export the fact that the fifth and sixth arguments of `possiblyInteractScheduled` are effects of actions `A` and `B`, and they may potentially interact; in `possiblyInteractScheduled` we check that they may actually overlap in time, considering temporal indeterminacy (at S) of the execution time of actions, which have not necessarily started, and of effects with respect to the actions.

For each of the modalities described earlier in this section and considered for a-posteriori conformance analysis, there are rules to define:

- necessary conditions for their applicability in a specific log, in order to prune the candidate explanations not supported by the log;
- how the CIG execution can be modified according to such modality.

Among the resulting answer sets, as in (Spiotta et al., 2015; Spiotta et al., 2017), optimization statements are used to select the answer set(s) with a minimum number of discrepancies with respect to the log. In the following section, we provide more details of our approach on specific examples.

6 A CONCRETE EXAMPLE

We consider the concurrent execution of a CIG for peptic ulcer (PU) and a CIG for venous thromboembolism (VT). 2 shows the two simplified CIGs at a high level of detail, using the GLARE representation. In the CIGs, the action “Amoxicillin therapy” (AT), belonging to PU, interacts with the action “Warfarin therapy” (WT, belonging to VT), which has the intention of avoiding the development of clots. Such an interaction is usually avoided in the medical practice, since it increases the anticoagulant effect of warfarin, raising the risk of bleedings. In 2 some temporal constraints are reported on delays between actions, and on action duration.

We applied our approach to three different logs for the two CIGs above. First, we describe a log in which no management has been applied, then we consider a log in which warfarin has been replaced with heparin, and finally we consider a situation in which the beginning of the warfarin therapy has been postponed until the end of the amoxicillin one. 3 represents the

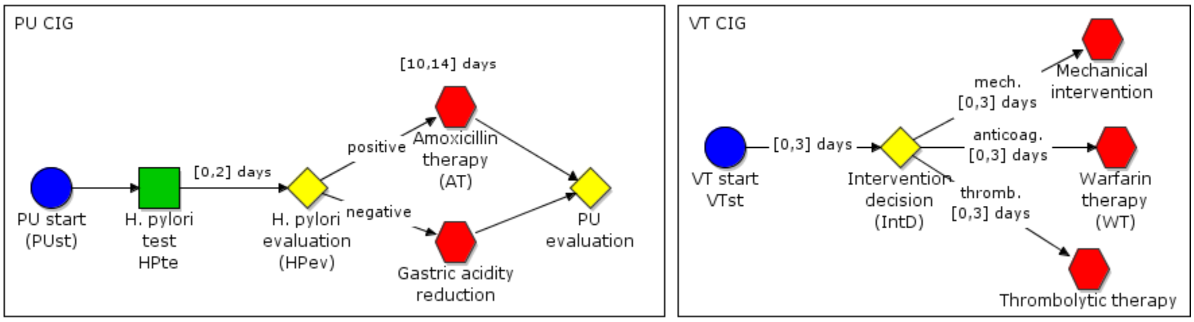

Figure 2: CIGs for peptic ulcer (PU) and venous thromboembolism (VT). Circles are atomic actions, hexagonal nodes are composite actions, diamond nodes are decisions.

three execution logs: the first row shows the log of PU (common to the three executions), while the rows 3-5 represents the different executions of VT. The second row represents the timeline, and an arrow from an action in a log to the timeline indicates that such an action has been executed (or started/ended) in that particular timepoint (e.g., action PUst has been executed at day 0). For durative actions, we indicate with `Act_s` and `Act_e` the starting and ending points of the action `Act`.

In all the examples, in the execution of the VT CIG, the anticoagulant therapy is selected (IntD); at the time of IntD, WT becomes scheduled, and its interaction is detected with AT, which is being executed (i.e., it is active).

Henceforth, we focus only on three options: `no_management`, `replanning` and `temporal_avoidance`. The most relevant rules for such options are described below. The following rule recognizes scenarios in which an interaction was possible and no management for it has been carried on.

```
info(possibly_interacting_actions_executed,A,B)
:-
possiblyInteract(Cg1,A,Cg2,B,_,_),
started(A,_),started(B,_),
{management(Cg1,A,Cg2,B,M): M<>no_management}0.
```

The first three conditions in the premise require that two actions `A` and `B`, which possibly interact considering temporal constraints, have started. The predicate `possiblyInteract` is analogous to `possiblyInteractScheduled`, except that it uses the actual execution time of the actions.

The following set of rules are relative to the replanning modality.

```
1: 1{substitute(Cg,A,C):hasEffect(C,E,_,_,_,_),
   causes(E,I),C<>A}1 :-
   management(Cg,A,_,_, replanning),
   aimsTo(Cg,A,I,I_s,I_e).
2: :- substitute(Cg,A,C), started(A,_).
3: :- substitute(Cg,A,C),
   hasEffect(C,E,E_si,E_sd,E_ed,E_ei),
   started(C,T_s), aimsTo(Cg,A,I,I_s,I_e),
   causes(E,I), ended(C,T_e),
   1{T_s+E_sd>I_s ; T_e+E_ed<I_e}.
4: block(A,S) :-
   substitute(_,A,_), relevantS(_,A,S).
5: succ(Cg,C,Anext) :-
   substitute(Cg,A,C), succ(Cg,A,Anext).
6: succ(Cg,Apred,C) :-
   substitute(Cg,A,C), succ(Cg,Apred,A).
```

The predicates `hasEffect(Act,E,E_si,E_sd,E_ed,E_ei)`, `causes(A,B)`, `aimsTo(Cg,A,I,I_s,I_e)` are exported from the knowledge server and model, respectively, the facts that an action `Act` has a particular effect `E`, starting between `E_si` and `E_sd` time units after `E`, and ending between `E_ed` and `E_ei` time units after `E`; that the effect/intention `A` causes `B`; that the action `A` in the CIG `Cg` has the intention `I` that must occur between `I_s` and `I_e`.

Basically, the first rule creates a candidate answer set for each possible action `C` having an effect achieving the intention of one of the interacting actions. Then, candidate answer sets considering the replacement of an action whose execution is reported in the log are discarded (rule 2). On the other hand, rule 3 discards candidates in which the replacing action `C` is executed in a time not compatible with the temporal constraints (if any) of the intention `I` of the original action `A` in the CIG. Finally, rules 4-6 replace the original action `A` with `C` in the CIG.

The following rule is relative to temporal avoidance.

```
:- management(Cg1,A,Cg2,B, temporal_avoidance),
1{not started(A,_); not started(B,_);
possiblyInteract(Cg1,A,Cg2,B,_,_)}.
```

The rule discards the temporal avoidance option if, in the execution log, one of the two actions has not been executed or they have been executed in times in which the interaction is still temporally possible. Further rules model the fact that, when justifying temporal avoidance, deviations from the temporal constraints in the CIGs are allowed.

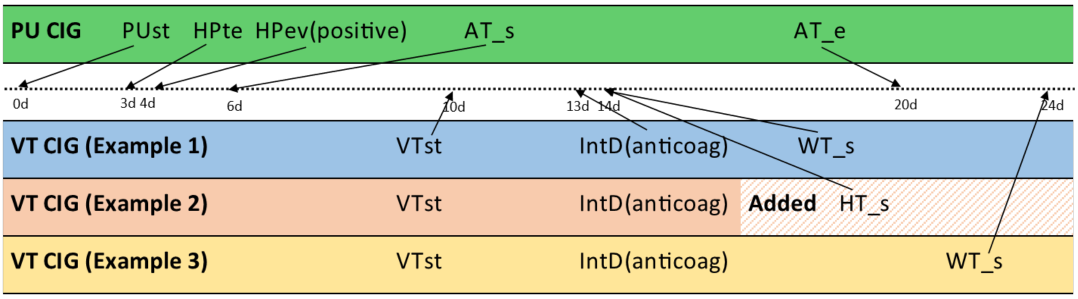

Figure 3: Graphic representation of the three execution logs considered. Repeated arcs for the actions VTst and IntD are drawn only for the first execution of VT.

Example 1. Consider the log of VT in the third row of 3, in which (i) WT starts during AT and (ii) no deviations with respect the original CIGs are present. Given (ii), replanning is ruled out, while (i) discards temporal avoidance. The only remaining alternative is `no_management`, which is reported as output together with the warning that a possible interaction could have occurred (i.e., `info(possibly_interacting_actions_executed, wt,at)`).

Example 2. The log in the fourth row of 3 shows an example in which "Heparin therapy" (HT), which is not present in the original CIGs, has been executed, while WT is not present. This last fact discards the option of temporal avoidance, while the no_management option does not match the log and is then discarded by minimization, since the log can be explained by the replanning option. In fact, in the ontological model of (Piovesan and Terenziani, 2015; Anselma et al., 2017), HT has the effect of decreasing the blood coagulation, which avoids the development of clots (i.e., the intention of WT). The resulting answer set contains the facts `management(vt,wt,pu,at, replanning)`, `substitute(vt,wt,ht)`.

Example 3. In the log in the last row of 3, the beginning of WT is delayed after the end of AT. In our knowledge model, the two actions interact due to the interaction between the "Anticoagulation" effect of WT and the "Platelet aggregation Inhibition" effect of AT, where the latter ends with the ending of AT. Thus, administering WT after AT_e is "safe" and the predicate `possiblyInteract` does not hold considering this log scenario, so that temporal avoidance is supported (and the resulting answer set contains the fact `management(vt,wt,pu,at, temporal_avoidance)`). On the other hand, replanning is not supported because WT and AT are present in the log, while no_management does not explain the fact that the log violates the temporal constraint stating a delay of `[0,3] days` between actions IntD and WT.

7 CONCLUSIONS

In this paper, we proposed the first approach that addresses the problem of analysing the conformance of execution traces with multiple CIGs, as needed in the treatment of comorbid patients. The importance of this task stems from the fact that full conformance to individual CIGs may be dangerous for comorbid patients: deviations are sometimes **necessary** to avoid undesired interactions between the CIGs. We thus identify cases in which traces are conformant, but undesirable interactions have not been avoided, and complement our conformance analysis with an explanation approach, aimed at justifying deviations in case they had avoided some possible undesired interaction. Additionally, two other main features of our approach, distinguishing it from the others in the literature, are:

(i) our attention to the temporal dimension and

(ii) our adoption of ASP to model and reason with the problem.

While in this paper we exploit the model of CIG action execution developed in (Spiotta et al., 2015; Spiotta et al., 2017), the rest of the approach is different. First of all, in (Spiotta et al., 2015; Spiotta et al., 2017) only one CIG is considered, and non-conformance can be explained on the basis of a "general" basic medical knowledge, which may trigger new actions in case problems not considered in the original CIG arise in the patient. In this paper two or more CIGs are considered, and more specific rules (the *modalities*) are used to explain non-conformances (in a context in which interactions are possible). Additionally, a central point of the current approach is the analysis of interactions between CIGs, while interactions (not even between the CIG and the actions suggested by the basic medical knowledge) were not taken into account in (Spiotta et al., 2015; Spiotta et al., 2017). As a consequence, the overall process of detecting and analysing non-conformances, as outlined in 1, is completely different from the conformance analysis carried on in (Spiotta et al., 2015; Spiotta et al., 2017). Some of the results of the analysis in (Spiotta et al., 2015; Spiotta et al., 2017) can be improved by the approach in this paper, which only considers deviations that can be justified by some form of interaction management. A more complete approach can however be obtained also considering general medical knowledge for dealing with relatively minor health problems whose treatment does not deserve developing a proper CIG, but can nevertheless be modeled in the same formal language as CIGs and possibly executed concurrently with the CIG actions. For this reason we plan, as a future work, to integrate the two approaches to provide a more comprehensive methodology. Moreover, we aim at demonstrating the coverage of the conformance approach in dealing with additional management modalities from (Piovesan and Terenziani, 2015).

Finally, we plan to experiment the approach also considering realistic traces containing also imprecise (and possibly missing) data, through a mixed-initiative approach.

ACKNOWLEDGMENTS

This research is original and has a financial support of the Università del Piemonte Orientale.

REFERENCES

Anselma, L., Piovesan, L., and Terenziani, P. (2017). Temporal detection and analysis of guideline interactions. *Artif. Intell. Med.*, 76:40–62.

Bottrighi, A., Chesani, F., Mello, P., Montali, M., Montani, S., and Terenziani, P. (2011). Conformance Checking of Executed Clinical Guidelines in Presence of Basic Medical Knowledge. In *BPM 2011*, pages 200–211. Springer.

Chesani, F., De Masellis, R., Di Francescomarino, C., Ghidini, C., Mello, P., Montali, M., and Tessaris, S. (2016). Abducing compliance of incomplete event logs. In *AI*IA 2016: Advances in Artificial Intelligence, LNCS 10037*, pages 208–222.

Fraccaro, P., Arguello Castelerio, M., Ainsworth, J., and Buchan, I. (2015). Adoption of Clinical Decision Support in Multimorbidity: A Systematic Review. *JMIR Medical Informatics*, 3(1).

Groot, P., Hommersom, A., Lucas, P. J. F., Merk, R.-J., ten Teije, A., van Harmelen, F., and Serban, R. (2009). Using model checking for critiquing based on clinical guidelines. *Artificial Intelligence in Medicine*, 46(1):19–36.

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

Merhej, E., Schockaert, S., McKelvey, T. G., and Cock, M. D. (2016). Generating conflict-free treatments for patients with comorbidity using answer set programming. In *KR4HC/ProHealth 2016, LNCS 10096*, pages 111–119.

Michalowski, M., Wilk, S., Michalowski, W., Lin, D., Farion, K., and Mohapatra, S. (2013). Using Constraint Logic Programming to Implement Iterative Actions and Numerical Measures during Mitigation of Concurrently Applied Clinical Practice Guidelines. In *AIME 2013*, number 7885 in LNCS, pages 17–22. Springer.

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

Piovesan, L., Molino, G., and Terenziani, P. (2015). Supporting multi-level user-driven detection of guideline interactions. In *HEALTHINF 2015*, pages 413–422.

Piovesan, L. and Terenziani, P. (2015). A Mixed-Initiative approach to the conciliation of Clinical Guidelines for comorbid patients. In *KR4HC 2015*, volume 9485 of *LNCS*, pages 95–108. Springer, Pavia.

Piovesan, L. and Terenziani, P. (2016). A Constraint-Based Approach for the Conciliation of Clinical Guidelines. In *IBERAMIA*, volume 10022 of *LNCS*, pages 77–88. Springer.

Quaglini, S. (2008). Compliance with clinical practice guidelines. In A. ten Teije, S. Miksch, P. L., editor, *Computer-based Medical Guidelines and Protocols: A Primer and Current Trends*. IOS Press.

Riaño, D. and Collado, A. (2013). Model-Based Combination of Treatments for the Management of Chronic Comorbid Patients. In *Artificial Intelligence in Medicine*, volume 7885, pages 11–16. Springer Berlin Heidelberg, Berlin, Heidelberg.

Sánchez-Garzón, I., Fernández-Olivares, J., Onainda, E., Milla, G., Jordán, J., and Castejón, P. (2013). A Multi-agent Planning Approach for the Generation of Personalized Treatment Plans of Comorbid Patients. In *AIME 2013*, number 7885 in LNCS, pages 23–27. Springer.

Spiotta, M., Terenziani, P., and Theseider Dupré, D. (2015). Answer Set Programming for Temporal Conformance Analysis of Clinical Guidelines Execution. In *KR4HC 2015*, volume 9485 of *LNCS*, pages 65–79. Springer.

Spiotta, M., Terenziani, P., and Theseider Dupré, D. (2017). Temporal conformance analysis and explanation of clinical guidelines execution: An answer set programming approach. *IEEE Trans. Knowl. Data Eng.*, 29(11):2567–2580.

Ten Teije, A., Miksch, S., and Lucas, P., editors (2008). *Computer-based medical guidelines and protocols: a primer and current trends*, volume 139 of *Studies in health technology and informatics*. IOS Press, Amsterdam.

Terenziani, P., Molino, G., and Torchio, M. (2001). A modular approach for representing and executing clinical guidelines. *Artif. Intell. Med.*, 23(3):249–276.

Wilk, S., Michalowski, M., Michalowski, W., Rosu, D., Carrier, M., and Kezadri-Hamiaz, M. (2017). Comprehensive mitigation framework for concurrent application of multiple clinical practice guidelines. *Journal of Biomedical Informatics*, 66:52–71.

Wilk, S., Michalowski, W., Michalowski, M., Farion, K., Hing, M. M., and Mohapatra, S. (2013). Mitigation of adverse interactions in pairs of clinical practice guidelines using constraint logic programming. *J Biomed Inform*, 46(2):341–353.

Zamborlini, V., da Silveira, M., Pruski, C., ten Teije, A., Geleijn, E., van der Leeden, M., Stuiver, M., and van Harmelen, F. (2017). Analyzing interactions on combining multiple clinical guidelines. *Artificial Intelligence in Medicine*.

Zamborlini, V., Silveira, M. D., Pruski, C., Teije, A. t., and Harmelen, F. v. (2014). Towards a Conceptual Model for Enhancing Reasoning About Clinical Guidelines - A Case-Study on Comorbidity. In *KR4HC*, volume 8903 of *LNCS*, pages 29–44. Springer.

Zhang, Y. and Zhang, Z. (2014). Preliminary Result on Finding Treatments for Patients with Comorbidity. In *KR4HC 2014*, number 8903 in LNCS, pages 14–28. Springer.