

# GLARE-SSCPM: an Intelligent System to Support the Treatment of Comorbid Patients<sup>1</sup>

Luca Piovesan and Paolo Terenziani, *Università del Piemonte Orientale*  
Gianpaolo Molino, *“San Giovanni Battista” Hospital*

**Abstract.** The development of software tools supporting physicians in the treatment of comorbid patients is a challenging goal and a “hot topic” in Medical Informatics and Artificial Intelligence. Computer Interpretable Guidelines (CIGs) are consolidated tools to support physicians with evidence-based recommendations in the treatment of patients affected by a specific disease. However, the applications of two or more CIGs on comorbid patients is critical, since dangerous interactions between actions from different CIGs may arise. GLARE-SSCPM is the *first* tool supporting, in an integrated way, (i) the knowledge-based detection of interactions, (ii) the management of the interactions, and (iii) the final “merge” of (part of) the CIGs operating on the patient. GLARE-SSCPM is characterized by being very supportive to physicians, providing them support for focusing, interaction detection, and for an “hypothesize and test” approach to manage the detected interactions. To achieve such goals, it provides advanced Artificial Intelligence techniques. Preliminary tests in the educational context, within the RoPHS project, have provided encouraging results.

**Keywords.** Medical expert systems, Medicine and science, Decision support, Knowledge based systems.

## Introduction

Comorbidities are, in the medical context, the co-occurrence in the same patient of two or more different diseases. In the last 20 years, Clinical Practice Guidelines (CPGs) have emerged in the Medical area as the major tool to promote *evidence-based* medicine, supporting quality, standardization, and optimization of the clinical treatments. CPGs are “*systematically developed statements to assist practitioner and patient decisions about appropriate health care in specific clinical circumstances*” [1]. Thousands of CPGs have been devised in the last few years. For instance, the Guideline International Network, which groups 103 organizations from all the continents, provides a library of more than 6000 CPGs. Recent research has shown that the development of computer-based approaches to acquire, represent, execute and reason with Computer-Interpretable Guidelines (CIGs henceforth) provides major advantages, and several CIG systems have been developed in the last twenty years, (see the surveys [2], [3]).

CPGs provide evidence-based information of interventions, but only on individual pathologies. The main problem with comorbid patients is that more than one CIG (one for each disease) are executed on them, and dangerous interactions between (the effects of) actions from different CIGs may arise. This sets up the need of developing ways of analysing and merging multiple single disease interventions to provide professionals' assistance to comorbid patients [4]. The development of these methods has been identified as one of the “grand challenges” for clinical decision support [5]. Since the early 2010's, the treatment of comorbidities has become a “hot” topic in the area of Medical Informatics, leading to innovative Artificial Intelligence architectures and methodologies (i) to detect interactions, or to (ii) manage them and “merge” the CIGs. However, GLARE-SSCPM (GLARE Support System for Comorbid Patient Management) is the first comprehensive framework supporting in an *integrated* and *physician-friendly* way both (i) and (ii) tasks.

---

<sup>1</sup> Published in IEEE Intelligent Systems. See <https://doi.org/10.1109/MIS.2018.2886697>

## GLARE-SSCPM: General Architecture and Behavior of the System

GLARE-SSCPM is an extension of GLARE (GuideLine Acquisition Representation and Execution, [6]), a well-known CIG framework. GLARE is the result of a long-term cooperation between the University of Eastern Piedmont and the San Giovanni hospital, started in 1997. The framework provides a formalism to represent CIGs, a tool to acquire them, a mechanism to execute a CIG on a determinate patient using patient's data. Both the acquisition and the execution processes are supported by a graphic interface (GUI). In particular, GLARE acquisition tool adopts different types of syntactic and semantic facilities to support physicians in the explication of medical knowledge in the guidelines [7]. In GLARE, CIGs are represented 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 distinguishes between *atomic actions* (simple steps in a CIG) and *composite actions* (plans), which are defined in terms of their components (thus supporting the definition of CIGs at different *levels of abstraction*). 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). GLARE also adopt advanced Artificial Intelligence techniques to support challenging tasks such as temporal reasoning, cost-benefit analysis (for decision support), model-based verification [8]. Additionally, GLARE also exploits a *Knowledge Manager*, i.e., a module coping with additional (CIG-independent) medical knowledge [9]. It adopts an OWL ontological model developed in collaboration with expert physicians, using Protégé as main instrument and integrating part of the SNOMED CT and ACT ontologies. Each action in GLARE can be associated with one or more elements of the ontological model describing, for instance, its effects or intentions (part of the model useful for this paper will be detailed in the following). Moreover, the Knowledge Manager module is provided with standard OWL reasoners providing inferences.

The first step in the extension of GLARE to cope with comorbidities was the identification of the different tasks to be solved. In strict cooperation with the physicians in our team, we identified the following tasks:

- (1) The detection of interactions occurring between CIGs
- (2) The management of the interactions
- (3) The final merging of the CIGs

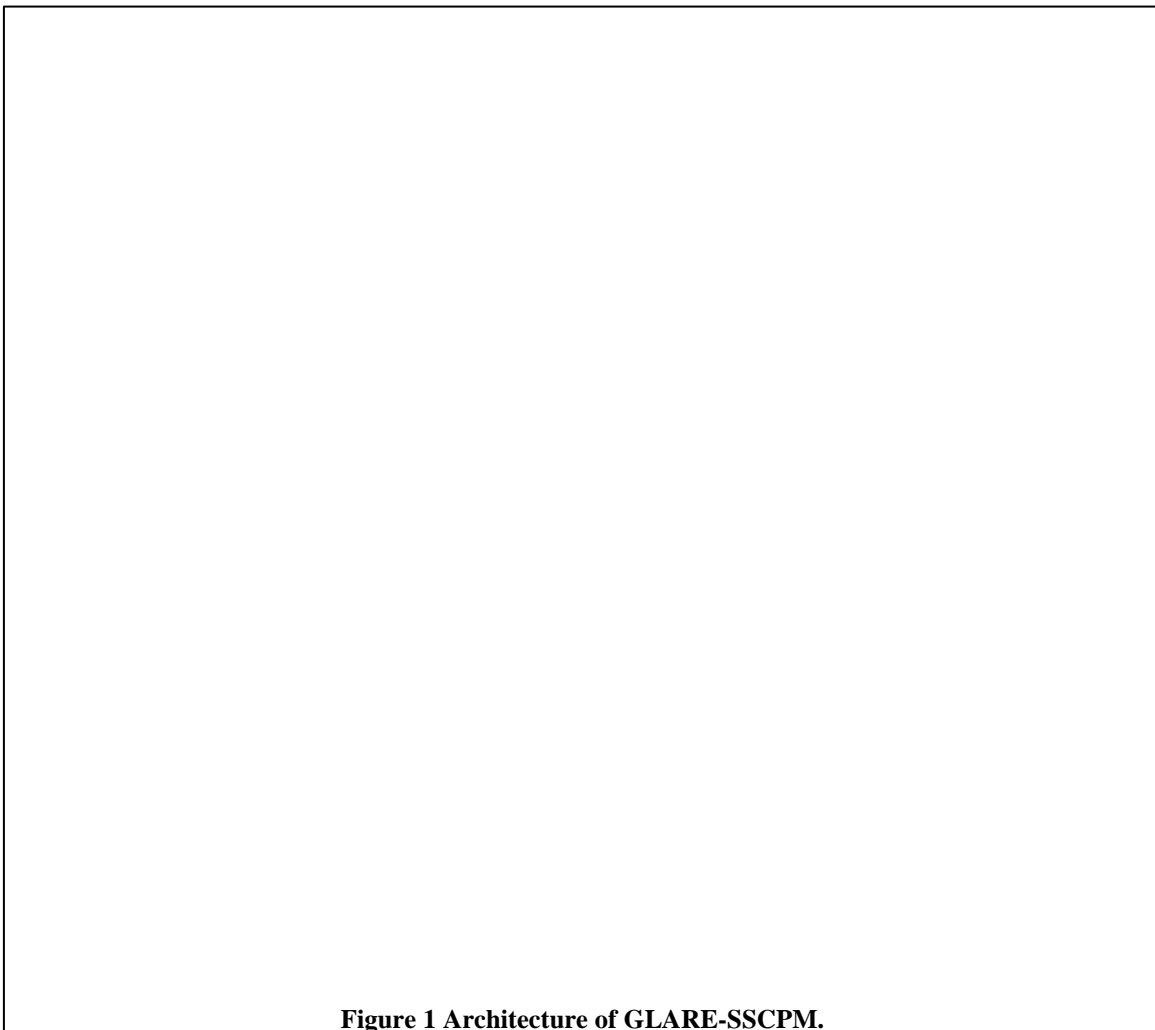
We have separately discussed in technical detail our techniques to cope with each task independently of the others in previous publications, mostly in conference papers (interaction detection [9], interaction management [10], conciliation [11] and temporal reasoning for comorbidities [12], [13]). Notably, to cope with the medical problem, several advances with respect to the AI state of the art were achieved, such as the treatment of complex – and not explored yet – temporal issues [13], and an innovative approach to conditional plan merging [11].

The second step, and first major contribution of this paper, is the definition of the intended *behaviour* of our overall approach, i.e., what are the facilities it should provide to physicians, and what is its intended interaction with them. As we will see in the comparison section, no current ICT approach copes with tasks 1-3 above together, and the large majority of them propose black-boxes tools. As regards interaction detection, such tools provide as output *all* the possible interactions between *all* the possible actions in the input CIGs. As regards management and merge (tasks 2 and 3 above), given a set of patient's data and CIGs to be executed, they automatically return a “merge” of the CIGs. Our 20-year experience of cooperation with physicians suggests that, although such approaches might present valuable Computer Science contributions, they are not likely to be used in the clinical practice. Indeed, physicians are the ultimate responsible of the health of their patients, and cannot delegate such a responsibility to a “black-box” fully automatic tool. ICT tools must *support* physicians, not taking their place. Together with physicians, we have identified a *flexible* interaction between ICT support and physicians, based on the “*focus, hypothesize and test*” modality below.

In our proposal, in a session of work on a comorbid patient, a physician can first (task 1; Interaction Detection) use GLARE-SSCPM to *focus* on the parts of the CIGs that are relevant to the current status of the patient, and use GLARE-SSCPM to find the possible interactions (if any) between the focused actions. A domain-independent knowledge base is used by GLARE-SSCPM, as well as temporal reasoning, to determine just those interactions that can effectively occur in time. Then (task 2; Interaction Management) a management for each interaction (if needed) can be chosen. GLARE-

SSCPM provides a wide set of alternative management options, and a way to physicians to simulate the effects of applying the chosen managements (“what if” analysis). Once physicians have chosen a set of managements (one for each “undesirable” interaction), GLARE-SSCPM provides them (task 3; CIG Conciliation) a way of automatically checking the consistency of the given solution, and a “local merge” of the CIGs, implementing all the management options. Notably, if there is an inconsistency (i.e., the “solution” is unfeasible) or if physicians want to check alternative solutions, GLARE-SSCPM supports them in a sort of “*hypothesize and test*” loop: new management options can be selected, and a new “local merge” can be obtained. The final solution is then a local adaptation of the CIGs, that can be executed by GLARE. Then, an outer loop (the “focusing” loop) is supported: physicians can move the focus (to the next part of the CIGs) and restart the process (interaction detection, management, and CIG “merge”).

Third, in this paper we first propose the overall architecture of GLARE-SSCPM (see Figure 1). Notably, there is a module for each one of the tasks below, plus a specialized temporal reasoner, and the communications between the different components implement the *focusing* and the *hypothesize and test* loops described above.



### **GLARE-SSCPM: Main Components**

**Running example.** We consider a comorbid patient affected by Venous Thrombosis (VT) and Peptic Ulcer (PU). In Figure 2.a, we show a screenshot of the GUI of GLARE-SSCPM during the interaction

analysis. Both the CIGs are simplified versions of the ones provided by the British National Institute for Health and Care Excellence ([www.nice.org.uk](http://www.nice.org.uk)). The upper rectangle in Figure 2.a shows the two original CIGs at the highest level of detail; the lower rectangle shows the expansion of H. pylori eradication (from PU) and of Pharmacological intervention (from VT), after a “focusing” action and the detection of the interactions, plus the management of one of them (see above). Figure 2.b shows the management of another interaction and 2.c shows the two CIGs after the final merging (for the sake of space, only a part of the GUI screenshots is shown).

### Interaction Detection

Usually, CIGs may consist of hundreds of nodes (representing actions). Thus, the physician’s knowledge is fundamental to focus the analysis only on the interactions that are relevant for the patient at hand. GLARE-SSCPM *Interaction Detection* module [9] provides flexible and interactive focusing tool allowing physicians to navigate through the different abstraction levels, thus supporting the natural methodology they adopt to cope with CIG analysis. Specifically, GLARE-SSCPM interaction detection algorithm maintains in a tree data structure, called *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  $\langle CIG_1, focus_1 \rangle$ ,  $\langle CIG_2, focus_2 \rangle$  determining the desired level of abstraction and the focused actions, and an interaction component, in



**Figure 2 Different phases of CIG merging with GLARE-SSCPM. Legend: blue nodes are work actions, green query actions, yellow decisions and red composite actions.**

which, for each pair  $\langle A_i, A_j \rangle$  of actions ( $A_i \in focus_1, A_j \in focus_2$ ), the detected interactions are pointed out. In our example, the physician decided, starting from the CIGs in the top of Figure 2.a (i.e., the root of the navigation tree), to focus on the composite action “helicobacter pylori eradication” (HPEr; in PU) and the composite action “pharmacological intervention” (PI; in VT) (i.e.,  $\langle CIG_1, focus_1 \rangle, \langle CIG_2, focus_2 \rangle$  instantiates to  $\langle PU, \{HPEr\} \rangle, \langle VT, \{PI\} \rangle$ ). Then, our system creates a node in the navigation tree, shown in the bottom of the figure, in which the focused actions HPEr and PI are expanded, accordingly to the hierarchical representation in GLARE’s PU and VT CIGs. Specifically, HPEr is expanded with the choice between clarithromycin and amoxicillin or metronidazole and amoxicillin, while the PI is expanded with the choice between warfarin and fondaparinux therapies.

Once the actions of interest are identified through focusing, physicians may perform interaction detection on them. The interaction detection is automatic: GLARE-SSCPM exploits the knowledge base and the OWL reasoner to retrieve all the interactions between the intentions, effects and drugs prescribed (in case of pharmacological actions) of the focused actions.

In our example, we suppose that the physician focuses on the detection of interactions between amoxicillin (two alternative occurrences, A1 and A2 in Fig.2a) and warfarin (W) or fondaparinux (F) (formally, the  $focus_1$  instantiates to  $\langle PU, \{A1, A2\} \rangle$  and  $focus_2$  to  $\langle VT, \{W, F\} \rangle$ ). S/he selects the actions from the GUI and asks to the system to perform interaction detection (i.e., the pairs  $\langle A_i, A_j \rangle$  of actions ( $A_i \in focus_1, A_j \in focus_2$  are:  $\{\langle A1, W \rangle, \langle A1, F \rangle, \langle A2, W \rangle, \langle A2, F \rangle\}$ ).

In short, the Knowledge Base (see Figure 1) contains the knowledge that amoxicillin has antiplatelet effects, and that warfarin has anticoagulant effects. Both effects cause an increase in the risk of bleedings. As a consequence, through the application of a set of Semantic Web Rules (SWRLs), the knowledge manager infers that an interaction between them is possible, leading to an increased risk of bleedings (see in the right part of the node, the interaction “W-A1:  $\langle ACTION: Warfarin therapy, SUBSTANCE: Warfarin, ACTION: Amoxicilin therapy, SUBSTANCE: Amoxicilin, CHANGES: risk of bleedings, MODALITY: increase \rangle$ ”). The detection of other interactions is similar.

## Interaction Management

Once detected and analysed, interactions must be managed. Management options are local changes to the CIGs, which make the original CIGs executable avoiding undesirable interactions and promoting desirable ones. Analysing the medical literature, we have identified the main *interaction management operations* adopted by physicians in the medical practice [10]:

**Safe Alternative:** avoid an interaction by proposing alternative paths in the CIGs.

**Replanning:** add new alternative actions with the same intentions of the original ones.

**Temporal Avoidance:** avoid an interaction by imposing additional temporal constraints.

**Effect Monitoring:** maintain the interaction, introducing a monitoring action to monitor the effects of the interaction itself.

**Dosage Adjustment:** mitigate an interaction by changing drug dosage.

**Interaction Mitigation:** mitigate an interaction through the introduction of a new action.

**Interaction Alignment:** force the occurrence of a desired interaction, by imposing that specific paths are executed at specific times in the two CIGs.

**Intention Alignment:** manage actions aiming to the same intention, by avoiding redundancy.

We defined the instantiation of the above management operations through the composition of three basic reasoning techniques:

- (1) *Backward Navigation* (NAV), navigating backward the CIGs to find alternative paths leading to or avoiding a given interaction
- (2) *Goal Based Planning* (GBP), that, given some intentions, interactively builds a CIG achieving such intentions
- (3) *Temporal reasoning* (TR), providing (through the Temporal Module) different temporal reasoning facilities, such as, e.g., when to execute actions to temporally avoid a given interaction.

Given an interaction, our *Interaction Management* module [10] provides physicians with the possibility of choosing the most appropriate management (option), and show how the original CIGs are modified by its application to the CIGs.

In our example, we suppose that the user-physician chooses to apply the safe alternative management option to the interactions “F-A1” and “F-A2”. The resulting constraints (blue-label elements) are shown in the rectangle “Managements” of the bottom element in Figure 2.a. Basically, they avoid the concurrent execution of the paths containing the amoxicillin therapy and the Fondaparinux one. For the interactions “W-A1” and “W-A2”, we suppose that the user-physician has chosen the replanning

management option. The system drove physicians, through a GUI, in the planning of a part of CIG having the same intention of the Warfarin therapy (i.e., to avoid thrombi). The result is a plan consisting of two composite actions, retrieved from the knowledge model, prescribing the acetylsalicylic acid and omeprazole therapies (see Figure 2.b). The system automatically added a decision before the warfarin therapy and, as other alternative, the acetylsalicylic acid therapy. Finally, it adds two constraints stating that amoxicillin and warfarin therapies cannot be executed together.

### **CIG Conciliation**

Once the interactions have been identified, and managed in isolation, the simple union of the original CIGs with the applied managements is not yet an executable CIG. Two aspects must be considered: (i) the management options lead to changes to the original CIGs that are “locally” consistent, but possibly not consistent with each other, and (ii) the conjunctive application of multiple management options can lead to the introduction of additional constraints. For these reasons, a final “merging” step is required. This is the goal of the *CIG Conciliation* module [11], which provides as output a “merged” CIG executable by GLARE.

We base our approach to CIG conciliation on the CSP (Constraint Satisfaction Problem) framework. The process is automatic, and it is divided into three phases. First, the system derives a CSP (CSPorig) from the focused part of the original CIGs, the execution log (where available) and the managements. After that, CSPorig is propagated. During the propagation process, the consistency of the problem is checked and new constraints are found. If CSPorig is consistent, a new (propagated) CSP is obtained, called CSPprop. Finally, our system transforms back it in a CIG executable in GLARE. Otherwise, if CSPorig is not consistent, an execution in which the interactions are managed as chosen is not possible and the user is driven in the changing of the chosen managements (going back to the management phase).

The result of the conciliation phase of our example is shown in Figure 2.c. The resulting CIG is consistent, but two additional constraints are added by the conciliation phase (red-label elements). In particular, the first one shows that in case helicobacter pylori eradication is required for PU, fondaparinux therapy cannot be executed. This constraint, even if obvious, is important in case physician has already started the anticoagulant therapy when helicobacter pylori test results positive: in this case, an exception is thrown and a further management is required. On the other hand, the second added constraint states that, in case the patient results positive to helicobacter pylori evaluation, the path of acetylsalicylic acid therapy must be chosen in the new decision added to the CIGs.

### **Temporal Module**

Temporal constraints are an essential ingredient of CIGs, and temporal reasoning is necessary for many tasks in the field of CIGs [14]. For instance, during interaction detection, temporal reasoning must be used to check whether possible interaction can actually occur or not, given the temporal information available in the knowledge base (temporal constraints between actions and the variations they cause) and in the CIGs (temporal constraints between actions), and the time of execution of previous actions on the patient. We have chosen to devote a specific module, the *Temporal module* [13] in Figure 1, operating as a *knowledge server*, to cope with temporal constraints and temporal reasoning. Our treatment of temporal constraints is grounded on the STP framework, that we have extended along several directions to support several facilities, such as checking whether two actions may temporally interact, or determining execution times of some future actions to avoid some interactions. As an example, the amoxicillin therapy for PU lasts 10-14 days. Suppose that the physician has started five days ago the amoxicillin treatment A1 and decides to apply temporal avoidance to avoid the A1-W interaction. Thus, the Temporal Module is invoked, to suggest a starting point for W that certainly avoids the overlap between the interacting effects of A1 and of W. In this simple example, the effects hold only during the action executions, thus, after temporal reasoning, the system suggests to start W after the end of A1. To certainly avoid the overlap, it considers the maximum duration of A1, suggesting to start W after 9 days from now.

## Related Work and Comparisons

In the last years, several AI approaches have been developed coping with the issue of managing comorbid patients (see the survey [15]). Previous approaches did either cope with (i) interaction detection, or with (ii) interaction management and CIG “merge”.

(i) **Interaction detection.** Few approaches focused on interaction detection. The most relevant approach in such an area is the one in [16]. It provides a CIG and domain-independent conceptual model for medical actions and reasoning forms operating on it. Moreover, in such a work general rules are proposed to identify different types of interactions on the basis of such a knowledge.

(ii) **interaction management and CIG “merge”.** Several approaches have been devoted to the generation of integrated CIGs. It is possible to distinguish between the approaches aiming at achieving “conservative” CIGs, and those that do not. The approach in [17], for instance, belongs to the latter category. It builds ad-hoc CIGs from scratch, using an agent-based approach. Agents with hierarchical planning capabilities represent experts in the treatment of specific diseases. The CIG coping with the comorbidity is obtained through the coordination of all the agents. On the other hand, most of the approaches in literature (included GLARE-SSCPM) adopt more conservative techniques, attempting to merge existing CIGs. We follow such a mainstream, since, in the real medical practice, physician need to follow as much as possible evidence-based recommendations, such as the ones proposed in the original CIGs. Within such approaches, the one in [18] uses constraint logic programming (CLP). A CLP is derived from the CIGs, the interactions and their managements. Finally, a mitigation algorithm is applied to obtain the merged CIG. Riaño and Collado [4] propose a model-based approach for the combination of treatments. They model treatments as oriented graphs composed by decisions and actions. With the help of physicians, they define a set of operators to merge decisions or actions. The combination of the original CIGs is obtained through the application of the operators.

In our opinion, GLARE-SSCPM advances the state of the art to several respects.

(1) **Integration.** GLARE-SSCPM is the first system that manages, in an integrated way, both aspects (i) and (ii) above.

(2) **User-friendliness and usability.** As discussed in the introduction, the main goal of GLARE and GLARE-SSCPM is to support physicians, not to substitute them. This aspect clearly distinguishes GLARE-SSCPM by the aforementioned approaches in the literature, which mostly work in a “black-box” fashion. In particular, current approaches are geared towards the detection of the management of *all* the possible interactions between CIGs. However, each CIG may consist of hundreds of actions and/or alternative paths of actions, and the treatment of each possible interaction (between each pair of actions) would lead to a quadratic number of interactions, most of which are totally useless for the treatment of the patient at hand, either because the physician knows that certain actions/path are not applicable to the specific patient, or because the interaction cannot occur *in time*. On the other hand, we propose focusing facilities, so that physicians can focus the detection and management only on the “relevant” parts of the CIGs. Additionally, as regards **CIG merge**, current approaches assume that a “table” with the possible interactions and their managements is defined a priori *by physicians*. Given such a “table”, they operate in a black-box fashion, directly providing as output the “merged” CIG managing *all* the interactions. This approach looks not acceptable for physicians, since the management of each given interaction usually depends on the specific patient and context of execution. A further improvement is that GLARE-SSCPM supports temporal constraints and temporal reasoning. We have already defined and published the different components of GLARE-SSCPM [8]–[11]. However, this paper first presents: (i) the definition of the behaviour of the whole system, i.e., of the way in which the system support physicians in a session of work on a comorbid patient, through a “*focusing, hypothesize and test*” interaction; (ii) the definition of an integrated architecture to achieve such a support. Notably, to achieve such goals, additional contributions have been provided, such as (iii) GLARE formalism and (iv) the Temporal module high-level formalism have been extended to be able to model also some constraints provided by the CIG conciliation module (i.e., by the CSP); (v) the CIG conciliation module has been modified to be able to operate also on limited parts of the input CIGs (i.e., on the focuses provided by the Interaction Detection module).

## Implementation and experimental evaluation

Starting from 1997, a prototype of GLARE has been implemented in JAVA, and extensively and successfully tested by the physicians of the San Giovanni Hospital to cope with different phenomena, including bladder cancer, reflux esophagitis, heart failure, and ischemic stroke. Since from one side

there is not a regional patient Electronic Medical Record and from the other side the hospitals have not financed yet a full engineered version of our system, GLARE has not achieved yet the status of “medical device”, so that it has not been used yet in the clinical practice. However, it has been extensively used for education of medical students and healthcare practitioners. GLARE-SSCPM has been prototypically implemented in Java and OWL. A preliminary version of GLARE-SSCPM has been already tested in the training of healthcare practitioners in the context of emergency medicine, to cope with poly-trauma, within the ROPHS (Report on the Piedmont Health System) project. The interactivity of GLARE-SSCPM and its capability of supporting “focus, hypothesize and test” sessions of work has been crucial in the success of our system, which has facilitated the learning process of a cohort of about twenty healthcare professionals. We foresee to provide more extensive evaluations in the education context, with a comparative analysis of the skills acquired with and without the adoption of GLARE-SSCPM. GLARE-SSCPM has also been tested by physicians in our group on a set of toy comorbidity examples, on simulated patients. Such preliminary tests have shown the our focusing and interaction detection facilities and “hypothesize and test” support actually facilitate physicians in the management of comorbidities. However, more extensive and systematic evaluations are needed, and are the goal of our future work.

## Conclusions

The treatment of comorbidities constitutes a challenging problem, since it involves the detection and management of the possible interactions between CIGs, and a final conciliation of the selected management options. In this paper, for the first time, we show the integrated architecture of GLARE-SSCPM, describing how different modules interacts to the overall goal of supporting physicians in the management of the above tasks, to cope with comorbid patients. GLARE-SSCPM constitutes a significant advance over the state of the art.

## References

- [1] Committee to Advise the Public Health Service on Clinical Practice Guidelines, Institute of Medicine, *Clinical practice guidelines directions for a new program*. Washington, D.C.: National Academy Press, 1990.
- [2] A. Ten Teije, S. Miksch, and P. Lucas, Eds., *Computer-based medical guidelines and protocols: a primer and current trends*, vol. 139. Amsterdam: IOS Press, 2008.
- [3] M. Peleg, “Computer-interpretable clinical guidelines: A methodological review,” *Journal of Biomedical Informatics*, vol. 46, no. 4, pp. 744–763, Aug. 2013.
- [4] D. Riaño and A. Collado, “Model-Based Combination of Treatments for the Management of Chronic Comorbid Patients,” in *Artificial Intelligence in Medicine*, vol. 7885: Springer Berlin Heidelberg, 2013, pp. 11–16.
- [5] D. F. Sittig *et al.*, “Grand challenges in clinical decision support,” *J Biomed Inform*, vol. 41, no. 2, pp. 387–392, 2008.
- [6] P. Terenziani, G. Molino, and M. Torchio, “A modular approach for representing and executing clinical guidelines,” *Artificial Intelligence in Medicine*, vol. 23, no. 3, pp. 249–276, 2001.
- [7] P. Terenziani, L. Anselma, A. Bottrighi, L. Giordano, and S. Montani, “Automatic checking of the correctness of clinical guidelines in GLARE,” *Stud Health Technol Inform*, vol. 129, Pt 1, pp. 807–811, 2007.
- [8] P. Terenziani, S. Montani, A. Bottrighi, G. Molino, and M. Torchio, “Applying artificial intelligence to clinical guidelines: the GLARE approach,” *Stud Health Technol Inform*, vol. 139, pp. 273–282, 2008.
- [9] L. Piovesan, G. Molino, and P. Terenziani, “Supporting Physicians in the Detection of the Interactions between Treatments of Co-Morbid Patients,” in *Healthcare Informatics and Analytics: Emerging Issues and Trends*, IGI Global, 2014, pp. 165–193.
- [10] L. Piovesan and P. Terenziani, “A Mixed-Initiative approach to the conciliation of Clinical Guidelines for comorbid patients,” in *KR4HC 2015*, vol. 9485, Pavia: Springer, 2015, pp. 95–108.
- [11] L. Piovesan and P. Terenziani, “A Constraint-Based Approach for the Conciliation of Clinical Guidelines,” in *Advances in Artificial Intelligence - IBERAMIA 2016*, 2016, vol. 10022, pp. 77–88.



- [12] L. Piovesan, L. Anselma, and P. Terenziani, “Temporal detection of guideline interactions,” in *HEALTHINF 2015*, 2015, pp. 40–50.
- [13] L. Anselma, L. Piovesan, and P. Terenziani, “Temporal detection and analysis of guideline interactions,” *Artif. Intell. Med.*, vol. 76, pp. 40–62, 2017.
- [14] P. Terenziani, E. German, and Y. Shahar, “The temporal aspects of clinical guidelines,” *Stud Health Technol Inform*, vol. 139, pp. 81–100, 2008.
- [15] P. Fraccaro, M. Arguello Castelerio, J. Ainsworth, and I. Buchan, “Adoption of Clinical Decision Support in Multimorbidity: A Systematic Review,” *JMIR Medical Informatics*, vol. 3, no. 1, p. e4, Jan. 2015.
- [16] V. Zamborlini, M. da Silveira, C. Pruski, A. ten Teije, and F. van Harmelen, “Towards a Conceptual Model for Enhancing Reasoning About Clinical Guidelines,” in *Knowledge Representation for Health Care*, vol. 8903, S. Miksch, D. Riaño, and A. ten Teije, Eds. Cham: Springer, 2014, pp. 29–44.
- [17] I. Sánchez-Garzón, J. Fernández-Olivares, E. Onaindía, G. Milla, J. Jordán, and P. Castejón, “A Multi-agent Planning Approach for the Generation of Personalized Treatment Plans of Comorbid Patients,” in *AIME 2013*, Springer, 2013, pp. 23–27.
- [18] S. Wilk, W. Michalowski, M. Michalowski, K. Farion, M. M. Hing, and S. Mohapatra, “Mitigation of adverse interactions in pairs of clinical practice guidelines using constraint logic programming,” *J Biomed Inform*, vol. 46, no. 2, pp. 341–353, 2013.

## The Authors

**Luca Piovesan** is a postdoctoral researcher at DISIT, Università del Piemonte Orientale since 2016. He obtained the PhD in Computer Science in 2016 at Università degli Studi di Torino. His research interests are knowledge representation, planning, decision support systems, temporal reasoning, medical informatics and temporal databases. Contact him at [luca.piovesan@uniupo.it](mailto:luca.piovesan@uniupo.it).

**Paolo Terenziani** is full professor in computer science with DISIT, Institute of Computer Science, Università del Piemonte Orientale. He received his PhD in computer science in 1993 from Università di Torino, Italy. His research interests include artificial intelligence (knowledge representation and temporal reasoning), databases and medical informatics. Contact him at [paolo.terenziani@uniupo.it](mailto:paolo.terenziani@uniupo.it).

**Gianpaolo Molino** (now retired) was the chief of the division of Internal Medicine, Center for Medical Informatics, Department of Internal Medicine at the San Giovanni Battista hospital. He received degrees in Medicine and Surgery, Gastroenterology, Occupational Medicine, Internal Medicine. His main research topics are hepatobiliary transports, cortico-hypophyseal function, compartmental models, expert systems, optimization of jaundice diagnosis, medical e-learning, medical informatics. Contact him at [gianpaolo.molino@virgilio.it](mailto:gianpaolo.molino@virgilio.it).