Concurrent Execution of Multiple Computer-interpretable Clinical Practice Guidelines and Their Interrelations

Eda BILICI[a,1], George DESPOTOU[a] and Theodoros N ARVANITIS[a]
[a] *Institute of Digital Healthcare, WMG, University of Warwick, Coventry CV4 7AL, UK*

Abstract. Execution of multiple computer-interpretable guidelines (CIGs), enables the creation of patient-centered care plans for multimorbidity, which can be monitored by clinical decision support systems. This paper introduces an execution framework to manage multiple, concurrently implemented CIGs, also discussing the approaches used such as constraint satisfaction.

Keywords. Computer-interpretable Guideline, Multimorbidity, Personal Care Plan, Model-Driven Engineering, Constraint Satisfaction; Model Transformation

1. Introduction

Clinical Practice Guidelines (CPGs) [1] are evidence-based statements, which are used to support carers in supplying appropriate care, mainly for patients with a single disease. Patients, especially the elderly, may have dynamic and multiple health conditions (multimorbidity) [2], which use multiple formalised versions of guidelines. A computer interpretable version of these guidelines (CIGs) [3] can be used to achieve automated connections between CPGs and patient data, in order to for supply error-free and consistent care recommendations to maintain patient safety. To date, several CIG-driven Clinical Decision Support Systems [4] have been proposed to acquire, represent and execute guidelines, using different guideline representation languages with associated execution engines (e.g., Asbru [5], GLIF3 [6], GLARE [7], Prof*orma* [8]). CIG execution involves instantiation of CPGs with patient data, using a mechanism to extract information, and recommend appropriate patient-specific care recommendations, such as care options and clinical information.

MuCIGREF, which is a Multiple Computer-interpretable Guideline Representation and Execution Framework, is developed to represent and execute CPGs and their associated knowledge constructs in order to generate personal care plans for multimorbid patients. The application process of each guideline follows the semantics of the MuCIGREF ontology. As the care application proceeds, the personal plan is updated according to the recommended actions of guidelines. However, this is a challenging issue when multiple guidelines are concurrently implemented [9]. These can be induced by managing a set of constraints relating with, for instance, arranging concurrency and synchronisation relations between clinical activities, recommended by the same or

[1] Corresponding Author, Eda Bilici, Institute of Digital Healthcare, WMG, University of Warwick, Coventry CV4 7AL, UK; E-mail: e.bilici@warwick.ac.uk.

different guidelines to avoid care conflicts (e.g., adverse drug interactions), or the need of multi-merging [10] of clinical activities to eliminate care duplications (e.g., inefficient use of resources). This paper presents the execution approach of MuCIGREF, mainly on alleviating challenges of parallel execution of multiple CIGs while generating a personalised care plan.

2. Model-driven Multiple Concurrently Implemented Guideline Execution

MuCIGREF implements the following three features:

1. **Elements of concepts and semantics regarding development of CIG models**: MuCIGREF uses the Eclipse Modeling Framework (EMF), which supplies a modelling and code generation architecture in Eclipse. EMF models are created to represent guideline elements and their interrelations. EMF-based CIG models are developed for each CPG. Once CIG models are created, they are executed in parallel to generate a unified personal plan model for each multimorbid patient. A model-to-model transformation approach (e.g., ATL [11]) is adopted, where source (individual CIG model) and target model (Personal Plan model) are complemented with a set of imperative logic, using the Epsilon Object Language (EOL) [12], a language imperative programming language to create, query and modify EMF models.
2. **Dynamic and flexible constraint satisfaction problems (CSPs) [13] over CIG models:** Actions of multiple guidelines may have diverse knowledge elements, their recommendations can be conflicting or overlapping, which may cause undesired patient outcomes. Constraints can be hard constraints, which must be satisfied at all times, like temporal constraints [14], or graph constraints to maintain care flow [15]; dynamic constraints like multi-activity management constraints to handle concurrency and synchronisation relations of guideline actions or to modify or optimise care to avoid conflicts or inefficiencies; or flexible constraints to handle user preferences. To do so, a new specialised CSP solving algorithm which is the extension of *backtracking* approach [13], is developed, which adopts to dynamic changes occurred in the CIG actions and their interrelations. Dynamic constraint satisfaction is used to support users to add new constraints (e.g., concurrency constraint), remove existing constraints or modify them during the solution process (e.g., care recommendation); flexible constraint satisfaction is used to relax constraints that must be satisfied, and enable users to make preferences on solutions (e.g., alternative care options).
3. **Consistent query answering [16] about these constraints:** Querying (e.g., specific time periods, patient information, lab results) and the constraint propagation technique [17], are used to reduce (filter) variables for the constraints, and extract information accordingly. Afterwards, this filtered information is recorded and used to update the model for further information extractions. Thus, step-by-step propagation and querying are applied in the entire personal plan generation process.

MuCIGREF's major execution functionalities are as follows: (i) checking execution status of each clinical activity to start care; (ii) identifying next clinical activities considering all the CIG models related with patient health disorders, by checking dependencies and constraints between clinical activities such as temporal constraints; and satisfaction of required conditions; (iii) adding, removing, or replacing clinical activities and/or their associated care elements; (iv) managing concurrency relations between multiple clinical activities to avoid harmful care advices, induced by recommendation conflicts (e.g., drug-drug interactions) and/or duplications (e.g., drug

overdose); (v) performing time-based synchronisation of clinical activities at the specific time point in order to be merged at the following care point as part of the care workflow; (vi) detecting unification care points to merge CIG actions; (vii) performing time-based care optimisation to avoid unnecessary resource (e.g., carer time, lab test) use and potential care duplications; and (viii) identifying conflicting clinical activities or potential conflicts and resolving them through modification of a clinical activity (e.g., activity start time, duration) or its associated care element (e.g., drug dose level).

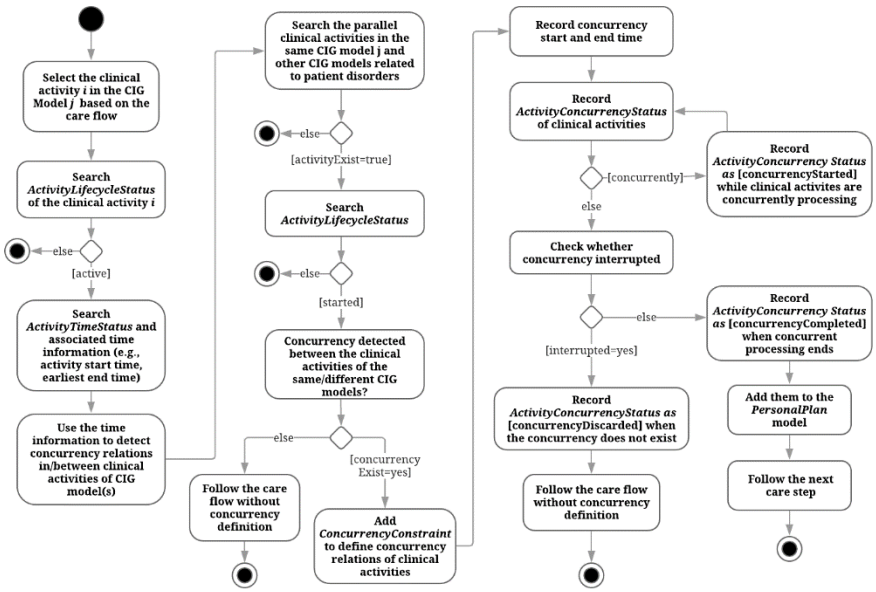

Figure 1. Concurrency management excerpt of MuCIGREF`s execution approach

3. Results

Several CPGs, from the UK National Institute of Care Excellence (NICE), are considered and, accordingly, personal plans are generated involving patient information. Execution framework meets the workflow requirements discussed in [18]. In Figure 1, an excerpt from the MuCIGREF's execution algorithm is presented, designed to handle concurrency relations between CIG actions. Validation is performed in the entire care process, in order to maintain consistent and error-free care plan. As part of the validation process, a set of model checking constraints are developed to comply with the requirements of execution regarding correctness, completeness, consistency and accuracy such as whether the defined care workflow has a cycle; each guideline has one starting activity and must have minimum one conclusion; or each decision must have minimum two conditional options. These constraints are applied in Epsilon Validation Language (EVL) [12] which check dependencies between the constraints specification of repairs that users can use to fix inconsistencies.

4. Discussion and Conclusion

In this work, conciliation of multiple CIGs with patient data to manage patients with multimorbidity, using a novel execution approach as part of the MuCIGREF, is introduced. Multimorbidity case studies were created with associated CPGs and patient data. Generating personal care plans for each patient by transforming individual CIG models; resolving challenges in coordination of complex knowledge sources and their interactions through satisfying a set of constraints with a new CSP solving approach; and adopting a dynamic model validation approach which supports users in each care step through supplying custom-built error messages, are the major contributions of this research. Future work will involve user validation and application in real-world cases.

Acknowledgments

EB has received a PhD scholarship from WMG, University of Warwick. GD and TNA have been supported by the EU H2020 C3-Cloud Project (under grant agreement No 68918) and Health Data Research UK (HDR-UK).

References

[1] M.J. Field and K.N. Lohr, *Clinical Practice Guideline: Directions for a new program*, National Academies Press, Washington, DC, 1990, 50.
[2] M.E. Salive, Multimorbidity in older adults, *Epiodemiol Rev* **35** (2013), 75-83.
[3] M. Peleg, Computer-interpretable clinical guidelines: A methodological review, *J Biomed Inform* **46** (2013), 744-763.
[4] E. Bilici, G. Despotou and T.N. Arvanitis, The use of computer-interpretable clinical guidelines to manage care complexities of patients with multimorbid conditions: A review, *Digital Health* **4** (2018), 1-21.
[5] A. Seyfang, S. Miksch and M. Marcos, Combining diagnosis and treatment using Asbru, *Int J Med Inform* **68** (2002), 49–57.
[6] A.A. Boxwala, M. Peleg, S. Tu et al., GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines, *J Biomed Inform* **37** (2004), 147–161.
[7] A. Bottrighi, G. Molino, S. Montani, et al., Supporting a distributed execution of clinical guidelines, *Comput Methods Programs Biomed* **112** (2013), 200-210.
[8] D.R. Sutton and J. Fox, The syntax and semantics of the PROforma guideline modeling language, *J Am Med Inform Assoc* **10** (2003), 433–443.
[9] S. Wilk, M. Michalowski, W. Michalowski et al., Comprehensive mitigation framework for concurrent application of multiple clinical practice guidelines, *J Biomed Inform* **66** (2017), 52-71.
[10] N. Mulyar, W.M. Van der Aalst and M. Peleg, A pattern-based analysis of clinical computer-interpretable guideline modeling languages, *J Am Med Inform Assoc* **14** (2007), 781-787.
[11] F. Jouault and I. Kurtev, Transforming models with ATL, *MODELS* (2005), 128-138.
[12] D. Kolovos, R. Louise, A. Garcia-Dominguez and R. Paige, *Epsilon Book*, 2013.
[13] I. Miguel and Q. Shen, Hard, flexible and dynamic constraint satisfaction, *The Knowledge Engineering Review* **14** (1999), 199-220.
[14] A. Kamišalić, R. David and W. Tatjana, Formalisation and acquisition of temporal knowledge for decision support in medical processes, *Comput Methods Programs Biomed* **158** (2018), 207-228.
[15] F. Orejas, H. Ehrig and U. Prange, A logic of graph constraints, *FASE* (2008), 179-198.
[16] M. Arenas, L. Bertossi and J. Chomicki, Consistent query answers in inconsistent databases, *PODS* **99** (1999), 68-79.
[17] C. Bessiere, Constraint propagation, *Foundations of Artif Intel* **2** (2006), 29-83.
[18] N.C. Russell, A.H.M. Ter Hofstede, W.M.P. Van Der Aalst and N.A. Mulyar, Workflow control-flow patterns: A revised view, *BPM Center Report BPM-06-22*, BPMcenter.org, 2006, 06-22.