Formal Modeling and Analysis of Home Care Plans

Kahina Gani, Marinette Bouet, Michel Schneider, and Farouk Toumani

LIMOS, CNRS, Blaise Pascal University, France
{gani,michel.schneider,ftoumani}@isima.fr,
marinette.bouet@univ-bpclermont.fr

Abstract. A home care plan defines all the services provided for a given patient at his/her own home and permits the coordination of the involved health care professionals. In this paper, we present a DSL (Domain specific language) based approach tailored to express home care plans using high level and user-oriented abstractions. Then we describe how home care plans, formalized as timed automata, can be automatically generated from these abstractions. We finally show how verification and monitoring of the resulting care plan can be handled using existing techniques and tools.

Keywords: Timed Automata, Domain Specific Language, Business Process Management, Home Care Plan, UPPAAL.

1 Introduction

A general trend that can be observed these recent years is to enable as much as possible patients to stay at their own homes instead of having long-term stays at hospitals or health establishments. This trend is motivated by obvious social and economic reasons. Several types of care may be provided at a patient's home including health services, specialized care such as parenteral nutrition or activities related to daily living such as bathing, dressing, toilet, etc. All the medical and social activities delivered for a given patient according to certain frequencies are scheduled in a so-called *care plan*. Hence, the notion of a care plan is a key concept in home care area. As part of the project Plas'O'Soins[1], we are interested by the problems underlying the design and management of home care plans.

The design of a care plan is however a difficult task. Indeed, process modeling in the medical field is not trivial because it requires complex coordination and interdisciplinary cooperation due to involvement of actors from various health care institutions [7]. Furthermore, care plans are essentially unstructured processes in the sense that each patient must have his/her own specific care plan. Therefore, it is simply not possible to design a unique process capturing in advance the care plans of all the patients. Another important feature of care plans lies in their associated complex temporal constraints. Indeed, the design of a

[1] http://plasosoins.univ-jfc.fr/

X. Franch et al. (Eds.): ICSOC 2014, LNCS 8831, pp. 494–501, 2014.

care plan requires the specification of the frequencies of the delivered home care activities. Such specifications are expressed by healthcare professionals in natural language, using usually a compact form: Every day morning, each Monday morning, etc. The home care activities are generally repetitive but may have irregularities or exceptions. Given the crucial role played by temporal constraints in home care plans, it appears clearly that such specifications could take benefit from existing theory and tools in the area of timed systems [3]. In this paper, we use *timed automata* [1], one of the most used modeling formalism to deal with timing constraints, as a basis to develop a formal framework to analyze care plans.

Solving the above problem and supporting design, analysis and verification, execution and monitoring of home care plan require tackling a number of challenges. The first challenge consists in the design and modeling of care plans. Due to the aforementioned features of care plans, it is not feasible to ask home care professionals to describe directly a care plan using a formal language such as, for example, timed automata. To cope with this difficulty, we first propose a DSL (Domain Specific Language) and a user centered specification language tailored to express home care plans using high level abstractions. We then define an automatic transformation of user specifications into timed automata. The resulting automaton is used to support automatic verification and monitoring of home care plans.

The paper is organized as follows. Section 2 describes the DSL based approach in which we mainly identify elementary temporal expressions. The general modeling process is presented at section 3 together with the construction of the proposed automata, i.e., pattern automata, activity automata and care plan automata. Section 4 presents some verification and monitoring issues. We discuss the result of this work at section 5.

2 A DSL-Based Approach for Specifying Home Care Plans

The design of a care plan is a complex collaborative process, managed by a primary medical coordinator and carried out by an interdisciplinary team. In order to understand such a design process and also to understand how a medical coordinator approaches the problem, we conducted in the context of the *Plas'O'Soins* project a thorough on-sites analysis of current practices in the field of home care.

This study showed the central role played by care plans as primary components of effective care coordination at patient's home. It appears therefore appropriate to provide tools to assist as much as possible the medical coordinator in the design of individual care plans as well as automated support for verification of the plan and monitoring of their executions. This is why in the Plas'O'Soins project, we propose a DSL based approach, tailored to express home care plans using high level abstractions.

A domain specific language (DSL) is a language designed to express a solution to a problem within a specific domain [10]. The proposed DSL provides high level abstractions that can be used by a medical coordinator to design a care plan for a given patient. The main building block in a care plan is the notion of activity. Our DSL includes several predefined activities identified by our analysis of the application domain. Each activity of the care plan is associated with a set of elementary temporal specifications. These specifications provide the information about the time when the activity should be performed, expressed as a quadruplet (Days, Time ranges, Period, Duration). In [6], we proposed a language that enables to express regular or irregular repetitions of an activity within some period in a condensed form, similar to that used by doctors. Figure 1

Predicted acts								
Act Id	Act	Temporalities			Duration	Main actor type	Nbr of actors	Int Id
		Days	Time ranges	Period				
A1	Toilet	Monday Wednesday Friday	Morning, evening	01/01/13-03/31/13	30	Nurse auxiliary	1	I1
		Sanday	Morning	01/01/13-03/31/13		Nurse auxiliary	1	
		...						
A2	Dress	Every day except(04/20)	Morning, evening	01/01/13-03/31/13	10	Nurse auxiliary	1	I1, I2
		04/20/13	Morning	01/01/13-03/31/13		Nurse auxiliary	1	
		...						

Fig. 1. Specification of activities Toilet and Dress

shows a simple example of a specification using this language. Each row of the table corresponds to an elementary temporal specification. In the quadruplet (Days, Time ranges, Period, Duration), Days and Time ranges fields can take different forms (patterns) to reflect the various possibilities encountered in the medical world [6]. Combination of elementary specifications permits to express superposition of different repetitions. Exceptions are introduced via the keyword except. Roughly speaking, the notion of a legal schedule of a care plan activity is defined as a sequence of allowed instances of this activity which satisfies the set of temporal specifications. An appropriate external representation of the care plan is crucial to facilitate the work of the coordinator. Figure 1 shows the current GUI (Graphical User Interface) developed to support a coordinator in designing a care plan using the proposed DSL.

3 General Modeling Process with Timed Automata

As said previously, we used formalism of timed automata (see for details [1]) to model home care plans. We consider in our work timed automata with ϵ-transitions (i.e., silent transitions) and *invariants* (i.e., guards on the states). The activities of care plans are not instantaneous but have a duration. This is why we need to capture the notion of duration of an activity in our timed automaton. This is achieved by considering three kinds of states: the *start states*, the *waiting states* and the *execution states*. More formally, a timed automaton used to model home care plans is defined as follows: $A = (S, s_0, \Sigma, X, Inv, T, F, W, E, St)$ where: S is a finite set of locations or states of the automaton with s_0 the initial state, $F \subseteq S$ is the set of final states, $W \subseteq S$ is the set of waiting states, $E \subseteq S$ is the set of execution states, and $St \subseteq S$ is the set of start states. Σ is a finite set of transition labels including $\{\epsilon\}$. X is a finite set of clocks. Inv: S $\rightarrow \phi$(X) associates an invariant to each state of the automaton and $T \subseteq S \times \Sigma \times \phi(X) \times 2^X \times S$ is a set of transitions.

As an example, Figure 2 shows the timed automaton corresponding to an activity A having a duration d. At the beginning the automaton is at the state $s_0 \in St$ then it starts the execution of the activity A when it enters the state $s_1 \in E$. The automaton stays at this state for the whole duration d of the activity A then it moves to the state $s_2 \in W$. The automaton uses the clocks $\{x_d, x_t\}$, invariants and transitions guards to control the execution of the activityA.

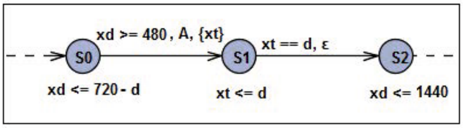

Fig. 2. Example of a timed automaton for a task A

We recall that our main objective is to build a care plan automaton for a given patient. To achieve this objective, we propose a three-steps approach which consists in: (i) mapping each elementary temporal specification into a pattern automaton, (ii) combination of pattern automaton to build an Activity automaton, and (iii) construction of global care plan automaton by composition of activity automata. These different steps are described below.

3.1 From Elementary Temporal Specifications to Pattern Automata

An elementary temporal specification is based on a temporal pattern chosen among several ones. We focus in this section on the case of the relative days pattern (the other patterns are described in [8]). Relative days pattern is used to express a regular repetition of the activity of the care plan. An example of relative days pattern can be found in the line 1 of the Figure 1. For each row of temporality defined for an activity a of the care plan, the corresponding timed automaton pattern $A_{RD} = (S, s_0, \Sigma, X, Inv, T, F, W, E, St)$ is defined as follows:

- S is a finite set of states, with s_0 the initial state. The total number of states is: NbStates $= 3 + ($NbTimeRanges - $1) * 2 *$ NbDays$+$ NbDays where NbTimeRanges is the number of times ranges and NbDays is the number of specified Days;
- F is the set of final states. We always have one final state;
- $\Sigma = \{$Activity name$\} \cup \{\epsilon\}$ is the set of transition labels;
- $X = \{x_d, x_t, x_p, x_w\}$ is the set of clocks, where x_d is used to control the execution of the activity within a day, x_t is used to control the activity duration, x_w is used to control the execution of the activity in a day of the week and x_p is used to control the execution of the activity in a day of the period. W.l.o.g., we assume that the time unit is the minute;
- $Inv = \{\forall s \in S, Inv(s) = (x_d \leq EndTimerange - d)$ and $s \in St, Inv(s) = (x_d \leq 24)$ and $s \in W, Inv(s) = (x_t \leq d)$ and $s \in E\}$;
- $T \subseteq S \times \Sigma \cup \{\epsilon\} \times \phi(X) \times 2^X \times S$ is the set of transitions. Each transition corresponds to a day of the week. The number of transitions is: NbTransitions $= 3 + (7 - NbDays) +$NbDays*2$+$NbTimeRanges-1 $* 2 * NbDays$

3.2 Activity Automata

This section gives the principles to construct an activity automaton by combining its associated patterns automata. This construction is illustrated on the example of the activity Toilet given at Table 1. Given a set of elementary temporal specifications of a given activity (expressed as rows of a table T), the corresponding activity automaton is built in the following steps:

Table 1. Elementary temporal specifications

Activity	Days	Time ranges	Period	Duration
Toilet	Monday Thursday	Morning	01/01/14-12/31/14	30
	Sunday	Evening	01/01/14-12/31/14	

Table 2. A modified table after step 1

Activity	Days	Time ranges	Period	Duration
Toilet	Monday Thursday	Morning	01/01/14-12/31/14	30
	Sunday	Evening	01/01/14-12/31/14	
	Everyday except(Monday Thursday Sunday)	None	01/01/14-12/31/14	

- Step 1: Add the following elementary temporal specification to T: Everyday except(specified Days) this specification is used to scan all days of the period.
- Step 2: Build pattern automaton for each elementary temporal specification except the last one (i.e., the one added in the previous step).
- Step 3: Build for each pattern automaton A, the corresponding special timed automaton $\hat{A}$. Informally, $\hat{A}$ recognizes timed words that encompass a timed word of A in sequences where the considered activity can also be executed anytime between two occurrences of this activity on the word of A.
- Step 4: Build for the added elementary temporal specification the corresponding special timed automaton $\tilde{C}$. $\tilde{C}$ recognizes timed words which enable the execution of the considered activity at anytime within the exception Days.
- Step 5: Build the intersection of special timed automata constructed at steps 3 and 4. The intersection is achieved following the classical construction defined in [2]. Figure 3 depicts the intersection automaton which encompasses all the possible schedules of the activity Toilet specified in Table 1.

3.3 Care Plan Automata

A care plan can be also described by means of a timed automaton which is obtained by composition of activity automata. For this purpose, we define a specific composition operator which mixes the asynchronous product (or shuffle) on some states and a specific synchronous product on other states (waiting states) in addition to blocking actions (in the execution states). Blocking is used to prevent the interleaving of activities in a care plan while synchronization is

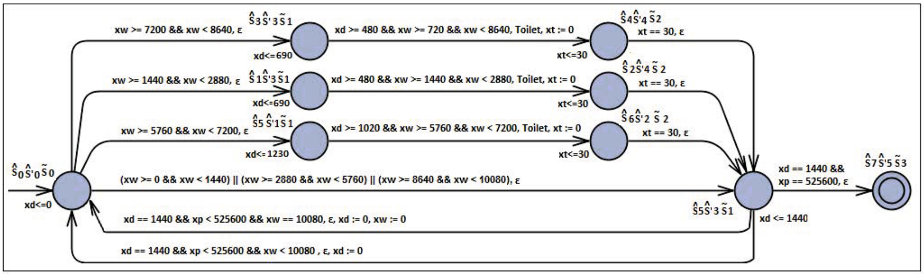

Fig. 3. Activity timed automaton (intersection of pattern automata)

needed when the activity automata are at waiting states in order to synchronize the reset of the day and week clocks (respectively, the variables x_d and x_w). In particular we propose to synchronize on ϵ -transitions (with reset) [5] when their origin states are waiting ones. We will see in what follows a more formal definition.

Definition 1. ***(Composition of timed automata)*** *Let $A_1 = (S_1, s_0^1, \Sigma_1, X_1, Inv_1, T_1, W_1, E_1, St_1)$ and $A_2 = (S_2, s_0^2, \Sigma_2, X_2, Inv_2, T_2, W_2, E_2, St_2)$ be two timed automata. The composition of A_1 and A_2, denoted $A_1 \times A_2$, is the timed automata $(S_1 \times S_2, s_1^0 \times s_2^0, \Sigma_1 \cup \Sigma_2, X s_1 \cup X_2, Inv, T)$, where Inv (S_1, S_2) = Inv $(S_1) \wedge$ Inv (S_2) and the transitions T is the union of the following sets:*

1. $\{((s_1, s_2), \epsilon, \phi, \lambda, (s_1', s_2'))$: $(s_1, \epsilon, \phi_1, \lambda_1, s_1') \in T_1$ *and* $(s_2, a, \phi_2, \lambda_2, s_2') \in T_2$, s_1 *and* s_2 *are both* $\in W\}$.
2. $\{((s_1, s_2), a, \phi, \lambda, (s_1', s_2'))$: $((s_1, a, \phi_1, \lambda_1, s_1') \in T_1, s_2 = s_2')$ *or* $((s_2, a, \phi_2, \lambda_2, s_2') \in T_2, s_1 = s_1')$, s_1 *and* s_2 *are both* $\in St\}$.
3. $\{((s_1, s_2), a, \phi, \lambda, (s_1', s_2'))$: $((s_1, a, \phi_1, \lambda_1, s_1') \in T_1, s_2 = s_2', s_2 \in W/St, s_1 \in E)$ *or* $((s_2, a, \phi_2, \lambda_2, s_2') \in T_2, s_1 = s_1', s_1 \in W/St, s_2 \in E)\}$.
4. $\{((s_1, s_2), a, \phi, \lambda, (s_1', s_2'))$: $((s_1, a, \phi_1, \lambda_1, s_1') \in T_1, s_2 = s_2', s_2 \in W, s_1 \in St)$ *or* $((s_2, a, \phi_2, \lambda_2, s_2') \in T_2, s_1 = s_1', s_1 \in W, s_2 \in St)\}$.

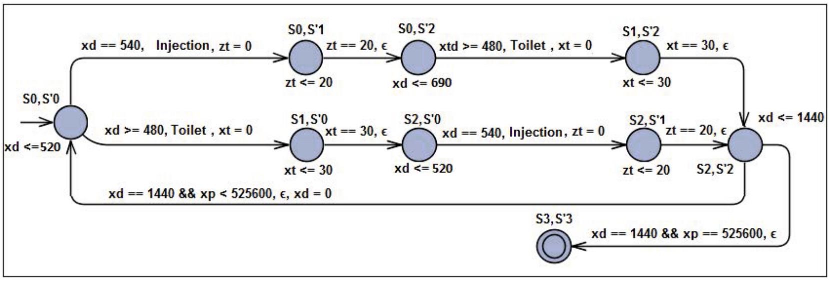

Fig. 4. Care plan timed automaton

Figure 4 shows the result of composition of the Toilet and Injection automata. The resulting automaton encompasses all the possible schedules of the activities Toilet and Injection.

4 Formal Analysis of Care Plans Using Timed Automata

With a formal model describing the behavior of care plans at hand, it becomes now possible to handle automatic verification and monitoring of care plans. We discuss below how to use the proposed framework to verify and monitor the home care plans using UPPAAL model checker [4].

Realizability of home care plans. It is important to check the realizability of a care plan, i.e., to check whether or not the activities included in the plan can be effectively scheduled and performed according to the constraints specified in the plan. In other words, a care plan is realizable when each activity can be performed without interruption in the imposed time range in any specified period. Checking realizability of a care plan can be reduced to the emptiness problem of the corresponding timed automaton.

Monitoring of home care plans. Note that most of the activities of a care plan are manual. In current state of affairs, the activities that have been performed are often recorded manually on paper. Our goal is to enable electronic recording of executed activities in order to keep track of the execution traces of care plans. Such information can then be used to monitor care plans. For example, compliance of executions traces w.r.t. a care plan may be checked by reducing this problem to the membership problem in the timed automata framework. Also, the monitoring system may be used to detect executions that deviate from the specification. More generally, a monitoring system can be enhanced with rules that enable to trigger alerts when particular deviations are detected.

Grouping activities into interventions. Grouping together activities that can be performed by a same type of actor (nurse, ...etc) and which occur in the same time range is called Intervention. The concept of intervention is really important in the sense that, it allows to reduce the waiting time between each activity in order to avoid multiple movings at the patient's home. The analysis of activities to specify interventions is a complex task since it requires to ensure compatibility of time ranges by taking into account the duration of each activity (multiple configurations are possible). It is also necessary to ensure that the grouped activities can be made by a same type of actor. The composition operator can be modified in order to incorporate the interventions in the computed car plan automaton (the obtained automaton is called interventions automaton). This is achieved by modifying the activity automaton in order to take into account the Intervention state. In addition, a specific clock variable, denoted Tmax, is added to control the *idle time* between the activities within the same intervention. In fact, the value of Tmax can be used as a parameter that can be defined by the coordinator and given as input to the composition operator to compute an interventions automaton. It is necessary to add an additional rule in the definition of the composition operator to take into account the new state intervention state. The rule is defined as follows: $\{((s_1, s_2), a, \phi, \lambda, (s_1', s_2'))$: $((s_1$, a, ϕ_1, λ_1, $s_1') \in T_1$, $s_2 = s_2'$, $s_2 \in W/E$, $s_1 \in Int)$ or $((s_2$, a, ϕ_2, λ_2, $s_2') \in T_2$, $s_1 = s_1'$, $s_1 \in W/St$, $s_2 \in Int\ \}$.

5 Discussion

We described in this paper, an approach to generate formal specifications of home care plans, expressed as timed automata, from a set of high level and user-oriented abstractions. We briefly discussed then how verification and monitoring of the resulting care plan can be handled using existing techniques and tools. The paper focuses on specific pattern (i.e., the relative days pattern). An extension of this work to the other patterns is described in [8].

Our specification language can easily be extended in order to increase its expressivity and usability. Extensions are performed by introducing other patterns for defining elementary temporal expressions. For example, patterns such as n times per day or per week, would be useful in a medical context.

In this study we considered only the activities of a single care plan relative to one patient. We intend to combine care plan automata to allow the planification of the interventions of several patients. It is necessary to take into account movements between patient homes and availability of human resources. Some works [9,11] have already highlighted the interest of automata for the activities planification. But in these works automata are directly designed by experts. In our approach, automata would result from high-level specifications produced by the administrator users.

References

1. Alur, R.: Timed automata. In: Halbwachs, N., Peled, D.A. (eds.) CAV 1999. LNCS, vol. 1633, pp. 8–22. Springer, Heidelberg (1999)
2. Alur, R., Dill, D.: A theory of timed automata. TCS (1994)
3. Alur, R., Henzinger, T.: Logics and models of real time: A survey. In: de Bakker, J.W., Huizing, C., de Roever, W.-P., Rozenberg, G. (eds.) REX 1991. LNCS, vol. 600, pp. 74–106. Springer, Heidelberg (1992)
4. Behrmann, G., David, A., Larsen, K.G.: A tutorial on UPPAAL. In: Bernardo, M., Corradini, F. (eds.) SFM-RT 2004. LNCS, vol. 3185, pp. 200–236. Springer, Heidelberg (2004)
5. Bérard, B., Petit, A., Diekert, V., Gastin, P.: Characterization of the expressive power of silent transitions in timed automata. Fundam. Inf. (1998)
6. Bouet, M., Gani, K., Schneider, M., Toumani, F.: A general model for specifying near periodic recurrent activities - application to home care activities. In: e-Health Networking, Applications Services (Healthcom) (2013)
7. Dadam, P., Reichertand, M., Kuhn, K.: Clinical workflows - the killer application for process-oriented information systems? Business (2000)
8. Gani, K., Bouet, M., Schneider, M., Toumani, F.: Modeling home care plan. Rapport de recherche RR-14-02, Limos, Clermont Ferrand, France (2014)
9. Abdeddaïm, Y., Maler, O.: Job-shop scheduling using timed automata. In: Berry, G., Comon, H., Finkel, A. (eds.) CAV 2001. LNCS, vol. 2102, pp. 478–492. Springer, Heidelberg (2001)
10. Menezes, A.L., Cirilo, C.E., de Moraes, J.L.C., de Souza, W.L., do Prado, A.F.: Using archetypes and domain specific languages on development of ubiquitous applications to pervasive healthcare. IEEE Computer Society (2010)
11. Paneka, S., Engella, S., Strsberg, O.: Scheduling and planning with timed automata. ISPSE, Elsevier (2006)