Time Management in Workflows with Loops

Margareta Ciglic(✉)

Department of Informatics Systems,
Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria
margareta.ciglic@aau.at
http://www.aau.at/tewi/inf/isys/ics/

Abstract. The goal of this research is to enable proactive time management for workflows with loops. We want to offer time constraint patterns that allow the formulation of time constraints on activities that are contained in loops.

Furthermore we design an algorithm for timed workflow graph computation considering loops and given time constraints. We use the time constraints to bind unbounded loops such that we iteratively expand the workflow, compute the timed workflow graph and check the satisfiability of the time constraints.

We also deal with a fast recomputation of a timed workflow graph at the runtime, which is needed to care for slack distribution, situation assessment and enactment of escalation strategies.

1 Introduction

Time management plays an important role in business processes, since violation of deadlines and other time constraints may lead to serious consequences. To represent temporal aspects, the workflow definition is extended by activity durations and other time constraints. Beneath overall deadlines, durations and implicit time constraints, we focus on two types of time constraints: the upper-bound (UBC) and the lower-bound constraint (LBC). They define the longest (respectively, shortest) time interval between the starting or ending points of two activities [4].

The aim of proactive time management is to predict and avoid violations of time constraints. A viable way to do that, as described in Eder et al. [4], is the computation of a timed workflow graph (TWfG) for a given workflow where each activity is annotated with execution intervals (earliest and latest finishing times) that are used to check the satisfiability of given time constraints.

An execution of a workflow and its time constraints is constantly monitored at the runtime and may be modified proactively if time constraints violations are predicted.

Time management becomes challenging if a workflow contains loops, especially if the loops are not bounded. Take a look at the simple photography workflow shown in figure 1. It contains 3 loops, denoted with *LS* (loop split) and *LJ*

Supervisor: Johann Eder, Alpen-Adria-Universität Klagenfurt, johann.eder@aau.at

I. Ciuciu et al. (Eds.): OTM 2015 Workshops, LNCS 9416, pp. 5–9, 2015.
DOI: 10.1007/978-3-319-26138-6_2

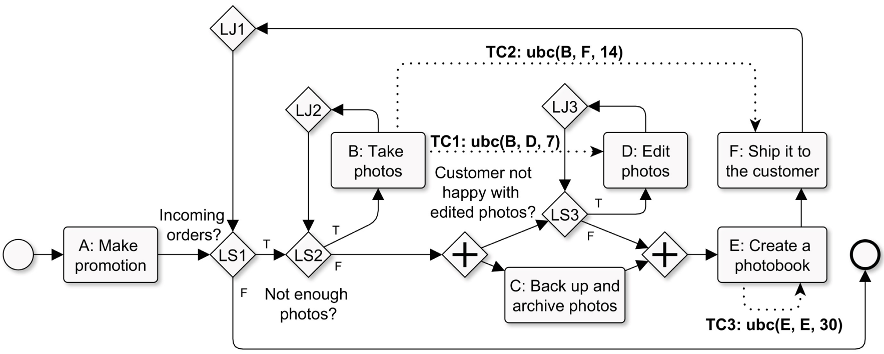

Fig. 1. Simple workflow example of a photography business

(loop join), that may iterate arbitrarily often. There are 3 time constraints for this workflow, defined by the customer and/or photographer:

TC1: ubc(B, D, 7) The customer wants to see the first edited photos at least 1 week after the shooting.

TC2: ubc(B, F, 14) According to the contract a photographer has to ship the photobook within 14 days.

TC3: ubc(E, E, 30) For each photobook a photographer gets a special price from the printing company if he/she orders it max. 30 days after the last order.

Each activity that is placed in a loop, might appear many times at runtime. Currently, there are no representations of constraints, which allow to specify which of these appearances of an activity are constrained. This limitation must be resolved with an extended definition of time constraints. One further problem is the satisfiability check of the time constraints, since the loops are not bounded and would have to be unfolded infinitely, which makes the calculation of the TWfG (and therefore also the satisfiability check of time constraints) impossible.

Loops, and problems related to them, do appear in workflows, therefore we tackle this challenge and search for a viable way to extend proactive time management to workflows with loops.

2 Related Work

The investigation of the research field workflow time management started back in the 90s. A short overview of the development of this field is given in [5]. An overview of time constraint types deliver Lanz et al. in [8] where they identify 10 classes of workflow time patterns. One of the categories of time patterns are recurrent process elements that also gives recommendations for specifying cyclic elements. First, Lanz et al. propose a general design choice for the number of cycles that a) can be determined by a fixed or dynamic number of iterations, b) depends on time lag and end date or c) depends on exit condition. A fixed number of iterations is often assumed in the literature, e.g. [3]. Lanz et al. further define that time lags between cycles can be fixed or may vary etc. The pattern solution for iteratively performed processes introduces a special time constraint between two process elements where the second one lies in the succeeding iteration.

Combi et al. [2] propose TNest, a new workflow modeling language for time constraints definition (among others), that can be used to express time constraints between two activities in different cycles of a loop, however the notation has a limited scope.

In the literature, loops are sometimes a) not handled at all [9], b) handled as a complex activity [10], c) rolled out into a sequence [12] or rolled out into conditional blocks (XORs) [7]. So far, most advanced loop handling was introduced by Pichler in [11]. He assigns branching probabilities to workflow graphs and uses this information to transform a cyclic workflow graph into an acyclic graph, called probabilistic unfolded workflow graph. To prevent an infinite growth of the graph, graph expansion stops when the probability of missing cases is below a certain threshold.

For the satisfiability check of time constraints, activities must be annotated with time information. There are several ways how to do that, e.g. Timed Petri Nets [6], Simple Temporal Network (STN) and its extended versions, Timed Game Automata (TGA) [1] or a timed workflow graph (TWfG) [4]. We use the TWfG to check the satisfiability of time constraints, because we believe that it is well suited for iterative graph expansion, computation of the time information and satisfiability check. TWfG computation works well for workflows without loops, however, it is not defined for workflows with loops.

As stated above, loops are frequently mentioned in the literature as a part of the workflow that is not the focus of the work and therefore they are not handled adequately [7,9,10,12]. Our contribution is to close this gap and to propose a time management solution for workflows with loops. We formulate time constraints definitions for workflows with loops and extend TWfG computation for adequate loop handling.

3 Research Goals

The challenge in proactive time management for workflows with loops can be divided into three main research goals:

1. Formulation of Time Constraints
The first problem is the definition of time constraints between two events where one or both of them appear in a loop. There is a need for an extended time constraint definition that can express the exact instance of an activity placed within a loop.

2. TWfG Computation and Time Constraints Satisfiability Check
Next problem is the TWfG computation in workflows with unbounded loops. A new algorithm that checks loop termination due to temporal constraints and the satisfiability of time constraints is needed.

3. Runtime Support
In order to keep the time information of the nodes in a TWfG accurate, a TWfG must be recomputed during the run time [4]. Since a recomputed TWfG must be quickly available at run time, an efficient recomputation algorithm that can cope with loops and status assessment are required and will be developed in scope of the thesis.

4 Work Plan and Research Methodology

According to the research goals, following work plan and solution ideas will be approached:

1. Formulation of Time Constraints
The first part of the solution copes with the definition of time constraints for workflows with loops. We extend the definition of the source and destination activity in an UBC/LBC such that we are able to address the exact appearance of an activity that is placed in a loop:

source := [FIRST | LAST | EACH] activity_label [WITHIN loop_label]

destination := [FIRST | LAST | EACH] [FOLLOWING | PRECEDING] activity_label [WITHIN loop_label [SAME_ITERATION | NEXT_ITERATION | PRECEDING_ITERATION]]

With the extended definition of UBC/LBC, we would express the 3 time constraints from the example in figure 1 as follows:

TC1: ubc(LAST B WITHIN $LS2$, FIRST FOLLOWING D WITHIN $LS1$ SAME_ITERATION, 7)
TC2: ubc(LAST B WITHIN $LS2$, FIRST FOLLOWING F WITHIN $LS1$ SAME_ITERATION, 14)
TC3: ubc(EACH E WITHIN $LS1$, FIRST PRECEDING E WITHIN $LS1$ PRECEDING_ITERATION, 30)

2. TWfG Computation and Time Constraints Satisfiability Check
We design an algorithm that is capable to iteratively expand a workflow, compute a TWfG and check the satisfiability of time constraints (inclusive the overall deadline). This approach allows the use of time constraints to bind unbounded loops and consequently enables proactive time management in workflows with loops.

3. Runtime Support
We design runtime support that enables fast TWfG recomputation, slack distribution and support for dispatchment, status assessment and enactment of escalation strategies.

In our work we generate two artifacts: the language for the representation of time constraints and the algorithm for TWfG computation and time constraints satisfiability checking. The evaluation is performed with formal proofs and a prototypical implementation of the algorithm.

References

1. Cimatti, A., Hunsberger, L., Micheli, A., Posenato, R., Roveri, M.: Sound and complete algorithms for checking the dynamic controllability of temporal networks with uncertainty, disjunction and observation. In: 21st International Symposium on Temporal Representation and Reasoning (TIME 2014), pp. 27–36. IEEE (2014)
2. Combi, C., Gambini, M., Migliorini, S., Posenato, R.: Representing business processes through a temporal data-centric workflow modeling language: An application to the management of clinical pathways. IEEE Transactions on Systems, Man, and Cybernetics: Systems **44**(9), 1182–1203 (2014)
3. Combi, C., Gozzi, M., Posenato, R., Pozzi, G.: Conceptual modeling of flexible temporal workflows. ACM Transactions on Autonomous and Adaptive Systems (TAAS) **7**(2), 19 (2012)
4. Eder, J., Panagos, E., Rabinovich, M.I.: Time constraints in workflow systems. In: Jarke, M., Oberweis, A. (eds.) CAiSE 1999. LNCS, vol. 1626, pp. 286–300. Springer, Heidelberg (1999)
5. Eder, J., Panagos, E., Rabinovich, M.: Workflow time management revisited. In: Bubenko, J., et al. (eds.) Seminal Contributions to Information Systems Engineering, pp. 207–213. Springer, Heidelberg (2013)
6. Foyo, P.M.G.D., Silva, J.R.: Using time petri nets for modeling and verification of timed constrained workflow systems. ABCM Symposium Series in Mechatronics **3**, 471–478 (2008)
7. Lanz, A., Posenato, R., Combi, C., Reichert, M.: Controllability of time-aware processes at run time. In: Meersman, R., Panetto, H., Dillon, T., Eder, J., Bellahsene, Z., Ritter, N., De Leenheer, P., Dou, D. (eds.) ODBASE 2013. LNCS, vol. 8185, pp. 39–56. Springer, Heidelberg (2013)
8. Lanz, A., Weber, B., Reichert, M.: Time patterns for process-aware information systems. Requirements Engineering **19**(2), 113–141 (2014)
9. Lu, R., Sadiq, S.W., Padmanabhan, V., Governatori, G.: Using a temporal constraint network for business process execution. In: Dobbie, G., Bailey, J. (eds.) ADC 2006 Proceedings of the 17th Australasian Database Conference. CRPIT, vol. 49, pp. 157–166. Australian Computer Society (2006)
10. Marjanovic, O.: Dynamic verification of temporal constraints in production workflows. In: 11th Australasian Database Conference (ADC 2000), pp. 74–81. IEEE (2000)
11. Pichler, H.: Time management for workflow systems. A probabilistic approach for basic and advanced control flow structures. Ph.D. thesis, Alpen-Adria-Universitaet Klagenfurt (2006)
12. Son, J.H., Kim, J.S., Kim, M.H.: Extracting the workflow critical path from the extended well-formed workflow schema. Journal of Computer and System Sciences **70**(1), 86–106 (2005)

http://www.springer.com/978-3-319-26137-9

On the Move to Meaningful Internet Systems: OTM 2015 Workshops
Confederated International Workshops: OTM Academy, OTM Industry Case Studies Program, EI2N, FBM, INBAST, ISDE, META4eS, and MSC 2015, Rhodes, Greece, October 26-30, 2015. Proceedings
Ciuciu, I.; Panetto, H.; Debruyne, C.; Aubry, A.; Bollen, P.; Valencia-García, R.; Mishra, A.; Fensel, A.; Ferri, F. (Eds.)
2015, XXIX, 578 p. 193 illus. in color., Softcover
ISBN: 978-3-319-26137-9