Checking Workflow Schemas with Time Constraints Using Timed Automata
(Extended Abstract)

Elisabetta De Maria, Angelo Montanari, and Marco Zantoni

Dipartimento di Matematica e Informatica, Università di Udine,
via delle Scienze 206, 33100 Udine, Italy
{demaria, montana, zantoni}@dimi.uniud.it

Abstract. Nowadays, the ability of providing an automated support to the management of business processes is widely recognized as a main competitive factor for companies. One of the most critical resources to deal with is time, but, unfortunately, the time management support offered by available workflow management systems is rather rudimentary. We focus our attention on the modeling and verification of workflows extended with time constraints. We propose (finite) *timed automata* as an effective tool to specify timed workflow schemas and to check their consistency. More precisely, we reduce the consistency problem for workflow schemas to the emptiness problem for timed automata, making it possible to exploit the machinery developed to solve the latter to address the former.

Workflow systems play an important role in the automation of business process management. They provide sophisticated tools for the specification and verification of the process structure (workflow schema), that allow one, for instance, to detect inconsistencies in process constraints and to identify process bottlenecks, as well as tools for monitoring and managing process execution, that make it possible, for instance, to trigger process-specific exception-handling activities when something wrong happens.

One of the most critical resources to deal with is time. Dealing with time constraints is crucial in designing and managing many business processes, and thus time management should be part of the core functionalities provided by workflow systems to control the lifecycle of processes. At build time, when workflow schemas are specified, workflow designers need means to represent time-related aspects of business processes, such as activity durations, time constraints between activities, deadlines, and timeouts, and to check their feasibility. At run time, when workflow schemas are instantiated and executed, process managers need mechanisms to detect possible time constraint violations and to trigger suitable exception-handling activities. Unfortunately, the time management support offered by available workflow management systems is rather rudimentary. The few existing approaches to time management in workflow systems are based on graph-theoretic techniques or on suitable refinements of Petri nets.

R. Meersman et al. (Eds.): OTM Workshops 2005, LNCS 3762, pp. 1–2, 2005.

We propose (finite) *timed automata* as an effective tool to specify workflow schemas with time constraints and to check their consistency [1]. More precisely, we reduce the consistency problem for timed workflow schemas to the emptiness problem for timed automata, making it possible to exploit the machinery developed to solve the latter to address the former. Furthermore, we take advantage of tools for timed automata to solve other relevant problems, such as, for instance, the problem of checking whether there exists an execution of a consistent workflow that satisfies some specific temporal properties.

A workflow is a collection of activities, agents, and dependencies between activities. Activities can be executed sequentially, repeatedly in a loop, or in parallel. Parallel executions can be unconditional (all activities are executed), conditional (only activities that satisfy a certain condition are executed) or alternative (one activity among several alternative ones is executed). In addition, workflows may contain optional activities (some activities may be executed or not). The control structure of the workflow implicitly defines a number of structural time constraints that basically states that an activity can start only when its predecessor activities have been completed. Explicit time constraints can be added to take into account time restrictions on activities imposed by organizational rules, laws, and commitments. The most common explicit time constraints are those on the duration of activity execution and those that constrain the delay between pairs of activities. Timed automata are one of the most widely used tools for the specification and verification of real-time systems. They are obtained from classical ones by adding a finite set of real-valued clock variables (clocks for short). Constraints on clocks are added to both automata states and transitions. Decidability of the emptiness problem for timed automata can be obtained by imposing suitable restrictions to automata structure and/or clocks. We take advantage of the decidability of deterministic timed automata with clock constraints only comparing clock values with constants.

We first show how basic workflow constructs for activity composition can be rendered in terms of the automata operations of concatenation, union, product, and intersection. Then, we show how the explicit time constraints of a workflow schema (activity duration, relative deadlines, upper/lower bound constraints) can be encoded into constraints on the finite set of real-valued clocks of a timed automaton. Putting together these two ingredients, we define a translation algorithm that maps a large set of timed workflow schemas into the above-mentioned class of deterministic timed automata. In such a way, we reduce the problem of consistency checking for timed workflow schemas to a reachability problem for timed automata: we have that the constraints of a given timed workflow schema are satisfiable (consistency problem) if and only if the language recognized by the corresponding automaton is not empty (reachability problem).

References

[1] E. De Maria, A. Montanari, and M. Zantoni. Checking workflow schemas with time constraints using timed automata. Technical Report UDMI/06/05, University of Udine, Mathematics and Computer Science Dept., 2005.