Mapping OWL-S Process Model to Timed Automata:
A Model-Checking Timed Temporal Logic Oriented Approach

Amel Boumaza, University Constantine II-Abdelhamid Mehri, Algeria
Ramdane Maamri, University Constantine II-Abdelhamid Mehri, Algeria

ABSTRACT
The conversion of web services to semantic web comes the opportunity to automate various tasks. OWL-S plays a key role in describing web services behaviour. While ontology-based semantics given to OWL-S is structural rather than behaviourally oriented, we cannot automate an essential task in this field, verification. In this article, the mapping of OWL-S process model to Timed automata is investigated, which is a suitable formalism for real time systems modeling and automatic verification. Hence, this has led to not only enabling automatic verification but also covering problems related to automated verification of temporal quantitative properties as bounded liveness property. As a starting point, the OWL-S and sub entry of time ontologies for describing the timed behaviour of services has been chosen. A defined set of mapping rules is used to automatically encode control constructs defined in OWL-S and temporal information into timed automata. Also, it is shown how a Uppaal checker is used to check required properties formulated in TCTL. Finally, an EClinic case study is used to illustrate the technique.

KEYWORDS
Formal Methods, OWL-S Ontology, Semantic Web, TCTL, Timed Automata, UPPAAL Tool, Web Services

INTRODUCTION
Service Oriented Architecture (SOA) provides standard for interoperating different software applications, running on heterogeneous platforms and/or frameworks. Actually, SOA has received a great deal of attention and is becoming an attractive solution for enterprise information systems.

Simultaneously, major effort has been undertaken to transform the actual Web, from distributed documents to a structure of interlinked resources, whose semantic is explicitly defined due to ontologies. Changing the way of using the Web has a clear impact on Web services. These services, which were formerly used through form-based Web pages, can now, be described in a machine-interpretable form enabling automation of many tasks currently performed by human-being. In literature, we can find a variety of ontology-based language for describing various aspects of Web

DOI: 10.4018/JITR.2018010103

Copyright © 2018, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
services: OWL-S, WSMO and SAWSDL are the most important approaches (Wang, Gibbins, Payne, Patelli, & Wang, 2015).

In the field of Web services, there is an increasing realization that temporal information plays a crucial role such as timeouts, durations, dates and times. The simplest way approach to specify time in semantic Web services is to simply use an ontology of time which can be able to help for managing time of real world services. However, this approach has several limitations: (1) It would be the programmer’s decision to encode manually temporal information and anchor them to events. Hence, there are many ways to encode the same scenario, that’s why the automation of the information retrieval and exploitation process often fails. For example, begin and end event’s time (a flight time) can be encoded as inputs parameters (arrival and departure times) but also can be encoded as temporal information anchored to events encoded as inputs parameters (arrival and departure events) (2) the type of semantics given to ontology-based work tends to be structural rather than behavioural oriented. As a result the semantics developed for OWL-S, are still deficient in automatic verification of Web services properties such as temporal ones and even for the simplest ones as deadlock free (3) complex time expressions may be specified by using temporal information (such as durations, timeouts,….) but it may leads to inconsistencies due to increasing complexity that’s why, it is important to provide ways to ensure correctness of behavior of a service with respect to relevant quantitative temporal properties. For example, arrival date should be at least one day before departure date in the case of a hotel booking, apartments can be rented only for periods longer than one month. The shipments must be delivered no later than 24 hours after the customer request.

In this paper, researchers address the limitations elucidated overhead through a novel approach to automate Web service verification of both temporal qualitative and quantitative properties for timed semantic Web services, which combines ontology of Web services, ontology of time and formal methods. Our approach covers both the specification of temporal information using an existed sub entry ontology of time (Pan & Hobbs, 2004) and formal verification through the mapping of the augmented specification of Web services by quantitative temporal information into timed automata. The contributions of this work are three-fold: First, researchers present generic pattern to better facilitate specification of temporal information on processes in OWL-S ontology and avoid ambiguity by adding new classes (i.e. process time and deadline classes). Second, researchers define a set of mapping rules to automatically encode various control constructs and temporal information defined in OWL-S into timed automata. The latter is equipped with a formal semantic which allow automatic verification of both kinds of temporal properties i.e. qualitative and quantitative properties. Third, researchers show how to use UPPAAAL checker for model checking required properties formulated in Timed Computation Tree Logic (TCTL) (Alur, Courcoubetis, & Dill, 1993).

The remainder of the paper is organized as follows. Section 2 highlights related work about proposed techniques and methods about formal verification of temporal properties on Web services. Section 3 sketches the models used in our approach namely OWL-S ontology and timed-automata. Section 4 defines the set of rules for automating translation from OWL-S description to timed automata network and show how it allows formal verification of temporal properties i.e. safety, liveness and bounded liveness property. Section 5 shows a case study analysis to demonstrate the use of our approach in applications. Finally, Section 6 concludes the work.

RELATED WORK

Although timed behavior specification and verification problem is well known by the formal methods community, previous works in the (semantic) Web services domain still insufficient. Authors in (Cambronerro, Díaz, Valero, & Martínez, 2011) present an approach for the validation and verification of composite Web services systems with timing restrictions. Web services are described in WS-CDL (Web Services Choreography Description Language). One of the main contributions of this paper is the definition of an operational semantics for a relevant subset of WS-CDL then a translation rules
Related Content

Network Implementation Project in the State Sector in Scotland: The Influence of Social and Organizational Factors
www.igi-global.com/chapter/network-implementation-project-state-sector/44629?camid=4v1a

The Effects of Computer-Mediated Communication on Inter-departmental Relationships: Propositions for Research
www.igi-global.com/article/effects-computer-mediated-communication-inter/50938?camid=4v1a
eInsurance Project: How to Develop Novel Electronic Services with Co-operation between Academics and Practitioners
Raija Järvinen, Jarno Salonen, Aki Ahonen and Jouni Kivistö-Rahnasto (2010). 
_Teaching Cases Collection_ (pp. 35-49).
[www.igi-global.com/article/einsurance-project-develop-novel-electronic/49195?camid=4v1a](www.igi-global.com/article/einsurance-project-develop-novel-electronic/49195?camid=4v1a)

A Practical Agile Framework for IT Service and Asset Management
ITSM/ITAM Through a Case Study
[www.igi-global.com/article/a-practical-agile-framework-for-it-service-and-asset-management-itsmitam-through-a-case-study/212625?camid=4v1a](www.igi-global.com/article/a-practical-agile-framework-for-it-service-and-asset-management-itsmitam-through-a-case-study/212625?camid=4v1a)