A Bounded Model Checking Approach for the Verification of Web Services Composition

Ehtesham Zahoor, National University of Computer and Emerging Sciences, Islamabad, Pakistan
Kashif Munir, National University of Computer and Emerging Sciences, Islamabad, Pakistan
Olivier Perrin, Laboratoire Lorrain de Recherche en Informatique et Ses Applications (LORIA), Université de Lorraine, Lorraine, France
Claude Godart, Laboratoire Lorrain de Recherche en Informatique et Ses Applications (LORIA), Université de Lorraine, Lorraine, France

ABSTRACT

In this paper, we propose a bounded model-checking based approach for the verification of declarative Web services composition processes using satisfiability solving (SAT). The need for the bounded model-checking approach stems from the nature of declarative processes as they are defined by only specifying the constraints that mark the boundary of the solution to the composition process. The proposed approach relies on using Event Calculus (EC) as the modeling formalism with a sound and complete EC to SAT encoding process. The use of EC as the modeling also formalism allows for a highly expressive approach for both the specification of composition model and for the specification of verification properties. Furthermore, as the conflict clauses returned by the SAT solver can be significantly large for complex processes and verification requirements, we propose a filtering criterion and defined patterns for identifying the clauses of interest for process verification.

Keywords: Declarative, Event-Calculus, Model-Checking, Verification, Web Services Composition

INTRODUCTION

Web services are defined to be the software systems that provide interoperable machine-to-machine interaction over a network. Individual services may need to be composed to form value added services and Web services composition has been highly active and widely studied research direction. The Web services composition process has different life-cycle stages. First, the process designer needs to model the composition process by using the fine-grained services to define new added-value processes. The objective of composition process modeling is to provide high-level specification independent from its implementation that should be easily understandable by the process modeler who creates the process, the developers

DOI: 10.4018/ijwsr.2013100103
responsible for implementing the process, and the business managers who monitor and manage the process. Then, the composition process needs to be verified to identify any anomalies and conflicts (such as deadlocks) in the process specification before execution. Furthermore, the composition process should be monitored, while in execution, to cater for unforeseen errors such as network failures.

A composition process model is termed as procedural when it contains explicit and complete information about the flow of the process. It only implicitly keeps track of why these design choices have been made and if they are indeed part of the requirements or merely assumed for specifying the process flow (Goedertier, 2008). In contrast, a declarative process model only requires specifying a set of constraints that mark the boundary of any solution to the composition process and any solution that respects these constraints is considered as a valid solution. Traditional approaches for modeling the Web services composition rely on a workflow-based approach where the process is modeled using approaches such as BPMN and is later translated to high-level languages such as WS-BPEL for its execution. While these approaches are intuitive and make it easier to model the processes, they are procedural in nature and they over-constrain the composition process, making it rigid and difficult to handle dynamically changing situations. In contrast, some declarative approaches have been proposed in the literature (Zahoor, Perrin, & Godart, 2010, July; van der Aalst & Pesic, 2006) that require the specification of constraints and are dynamic and flexible in nature.

An important feature of procedural approaches is that they provide well-known methods for verifying soundness of the composition, and to define strict semantics associated to the processes. Specifying the exact and complete sequence of activities to be performed for the composition process, as required by the traditional procedural approaches, does make it possible to use proposed automata or Petri nets based approaches for design-time verification of composition process, as the state-space is already known and relatively small. In contrast, the state-space of a declarative process can be significantly large, as the process is only partially defined and as all the transitions have not been explicitly defined. This makes it difficult to use traditional approaches for the verification of declarative Web services composition processes. In this paper, we propose an approach for the verification of declarative Web services composition processes using satisfiability solving. The proposed approach allows for identification of conflicts, hard constraints and inconsistencies in process specification and also allows resolving the identified conflicts. Specifically, our contributions include:

- **A Bounded Model Checking Approach:** For a declarative composition process, the state space can be significantly large, as all the transitions are not explicitly defined. State-space explosion would indeed be a challenge for complex and partially defined processes. We propose a SAT based bounded model checking approach, which does not require exponential space and is very efficient as the state space is searched in an arbitrary order.

- **Event-Calculus (EC) Based Approach:** For the bounded model checking approach, it is first required to encode the sequential behavior of a transition system over a finite interval as a propositional formula. The encoded formula is then given to a propositional decision procedure, satisfiability (SAT) solver. The proposed approach relies on using EC as the modeling formalism with a sound and complete EC to SAT encoding process (Mueller, 2004). It also provides implementation support (DECReasoner) to directly convert the EC based models to SAT problem and to invoke a SAT-solver.

- **Expressive Approach:** The use of EC as the modeling formalism allows for a highly expressive approach for both the specification of composition model and for the specification of verification properties. Verification properties can be specified for
Usage Profiles: A Process for Discovering Usage Patterns over Web Services and its Application to Service Evolution
www.igi-global.com/article/usage-profiles/86260?camid=4v1a