Data & Knowledge Engineering 87 (2013) 167–184

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Diagnosing correctness of semantic workflow models

Diana Borrego [a,*], Rik Eshuis [b], María Teresa Gómez-López [a], Rafael M. Gasca [a]

[a] University of Seville, Department of Computer Languages and Systems, Av Reina Mercedes S/N, 41012 Seville, Spain
[b] Eindhoven University of Technology, School of Industrial Engineering, P.O. Box 513, 5600 MB Eindhoven, The Netherlands

ARTICLE INFO

Article history:
Received 17 December 2011
Received in revised form 17 April 2013
Accepted 17 April 2013
Available online 25 April 2013

Keywords:
Workflow
Business process management
Diagnosis
Constraint programming
Integer programming

ABSTRACT

To model operational business processes in an accurate way, workflow models need to reference both the control flow and dataflow perspectives. Checking the correctness of such workflow models and giving precise feedback in case of errors is challenging due to the interplay between these different perspectives. In this paper, we propose a fully automated approach for diagnosing correctness of semantic workflow models in which the semantics of activities are specified with pre and postconditions. The control flow and dataflow perspectives of a semantic workflow are modeled in an integrated way using Artificial Intelligence techniques (Integer Programming and Constraint Programming). The approach has been implemented in the DiagFlow tool, which reads and diagnoses annotated XPDL models, using a state-of-the-art constraint solver as back end. Using this novel approach, complex semantic workflow models can be verified and diagnosed in an efficient way.

1. Introduction

Nowadays, organizations automate their business processes with workflow models that can be enacted using workflow management systems (WFMSs). For organizations it is essential to ensure the correct operation of workflow models at design time, before the workflow models get enacted. An incorrect operational workflow can dissatisfy customers and fixing the errors can be very costly, certainly compared to the costs of fixing the workflow model before it is deployed. Correctness of a workflow model can be verified by exhaustively checking all possible executions. Detected errors should be *diagnosed*, for instance by providing an error path that shows the cause of the error, such that errors can be repaired in a quick and effective way [1].

Workflow models can reference different perspectives [2]. Most workflow modeling and verification approaches only consider the control flow perspective [1,3–10], which is about the order in which the individual activities of a business process are executed. Another relevant perspective is the dataflow perspective [11], which details the flow of data among activities subject to certain constraints. The dataflow perspective is important because data constraints influence the possible executions of activities [11] and in turn, the execution of activities results in certain data constraints being enforced.

An effective means to express data constraints is to annotate activities in a workflow model with pre and postconditions that specify the effect on the data state for each activity. For instance, in the Sarbanes–Oxley Act of 2002, the internal audit department takes the lead and works alongside workflow owners for each process that has a direct effect on the data for the financial reporting. Annotating activities inside these processes with pre and postconditions facilitates compliance checking to ensure that workflows are properly designed.

Only recently, approaches for verifying workflow models with dataflows have been proposed [11–15]. However, these approaches do not consider diagnosis of dataflow errors. Diagnosing dataflow errors is complex due to the interplay between control flow and dataflow dependencies, as we explain in Section 2 with an example.

* Corresponding author. Tel.: +34 954 556 234; fax: +34 954 557 139.
E-mail addresses: dianabn@us.es (D. Borrego), h.eshuis@tue.nl (R. Eshuis), maytegomez@us.es (M.T. Gómez-López), gasca@us.es (R.M. Gasca).

0169-023X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.datak.2013.04.008

The goal of this paper is to develop an approach for *diagnosing the correctness of semantic workflow models*, containing activities whose effects are formally specified using pre and postconditions. An activity can start if the execution of the workflow model has reached the activity and its precondition is satisfied. Upon completion, the activity delivers data that satisfies its postcondition. An execution of the workflow can reach an activity whose precondition is not satisfied. In that case the execution gets stuck at the activity and fails.

We distinguish between two different notions of correctness to diagnose such dataflow errors:

- May-correctness. A workflow model is may-correct if every activity can be executed at least once, so there is an execution in which the activity is done.
- Must-correctness. A workflow model is must-correct if every possible execution that reaches an activity satisfies the precondition of the activity.

The diagnosis is performed at design-time, using Artificial Intelligence techniques to compute the execution instances allowed by a workflow model. For diagnosis, the workflow model is translated into two models: (1) an Integer Programming model (IP model), to determine the different instances of execution of the workflow, and (2) the preconditions and postconditions of the activities are modeled as constraints in a Constraint Satisfaction Problem (CSP) [16], following a BNF grammar in order to avoid any ambiguity.

This paper makes several contributions:

- *Workflow data graphs* are proposed as a formalism for modeling semantic workflows with pre and postconditions for the activities. These conditions are modeled as constraints according to a well-defined grammar in BNF.
- Two correctness notions for workflow data graphs, *may* and *must-correctness*, are proposed and novel diagnosis algorithms are developed for verifying may and must-correctness. The algorithms are complete: neither false positives nor false negatives are generated. Moreover, the algorithms offer precise diagnosis of the detected errors, indicating the execution causing the error where the workflow gets stuck.
- The approach has been implemented in the DiagFlow tool, presented in [1]. The tool reads XPDL models [17] in which the semantics of activities and the corresponding dataflow are specified using extended attributes.

This paper is organized as follows. Section 2 presents a motivating example to illustrate the concepts of may and must-correctness. Section 3 introduces workflow data graphs as a formal model for semantic workflows and defines may and must-correctness on workflow data graphs. Section 4 defines the IP and CSP formulations of a workflow data graph. The process of diagnosis is explained, and two algorithms are presented. The diagnosis of the motivating example is performed. Section 5 gives implementation details. Section 6 shows experimental results. Section 7 presents an overview of related work found in the literature. And finally, conclusions are drawn and future work is proposed in Section 8.

2. A motivating example

This section introduces an example of a semantic workflow model, shown in Fig. 1. This example describes the handling of a conference for an organizing committee, and it is used to illustrate the concepts of may and must-correctness in semantic workflow models. We use BPMN 2.0 [18] to visualize workflow models.

Fig. 1 shows a workflow that consists of nine activities (rectangles with rounded corners) and eight gateways or control nodes (diamonds), and a start and end event (circles). A gateway with one incoming edge and multiple outgoing edges is called a split; a gateway with multiple incoming edges and one outgoing edge is a join. Gateways with the +-symbol are AND: all incoming edges are required to pass the gateway, and the gateway activates all outgoing edges. The other gateways are XOR: one incoming edge can pass the gateway and one of the outgoing edges is activated as a result. In the figure, the activity labels are abbreviations of activity names that are listed in Table 1. The workflow performs the following steps:

1. The workflow starts with the establishment of the conference rate (ECR activity), in order to begin the registration period.
2. In the activity SAP, the process of acceptance of papers for the conference takes place. The number of final papers is determined.
3. The workflow is split into two branches. In the upper one, the cost of the gala dinner (D) and the lunches (L) to serve during the conference are calculated concurrently. On the lower branch, the workflow is routed according to the money spent in the social events during the conference (O or OS activities).

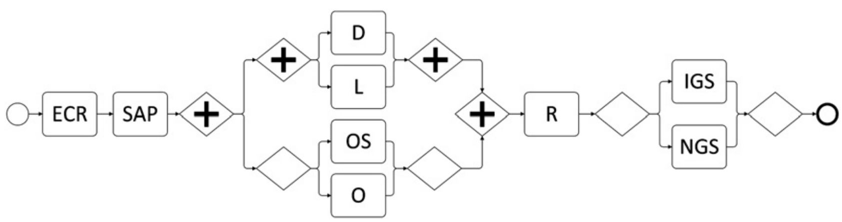

Fig. 1. Motivating example.