A Method for Assessing Parameter Impact on Control-Flow Discovery Algorithms

Joel Ribeiro[1] and Josep Carmona[1]

Universitat Politècnica de Catalunya, Spain.
{jribeiro, jcarmona}@cs.upc.edu

Abstract. Given an event log L, a control-flow discovery algorithm f, and a quality metric m, this paper faces the following problem: what are the parameters in f that mostly influence its application in terms of m when applied to L? This paper proposes a method to solve this problem, based on *sensitivity analysis*, a theory which has been successfully applied in other areas. Clearly, a satisfactory solution to this problem will be crucial to bridge the gap between process discovery algorithms and final users. Additionally, recommendation techniques and meta-techniques like determining the *representational bias* of an algorithm may benefit from solutions to the problem considered in this paper. The method has been evaluated over a set of logs and the flexible heuristic miner, and the preliminary results witness the applicability of the general framework described in this paper.

1 Introduction

Control-flow discovery is considered as one of the crucial features of Process Mining [13]. Intuitively, discovering the control-flow of a process requires to analyze its executions and extract the causality relations between activities which, taken together, illustrate the structure and ordering of the process under consideration.

There are many factors that may hamper the applicability of a control-flow discovery algorithm. On the one hand, the log characteristics may induce the use of particular algorithms, e.g., in the presence of *noise* in the log it may be advisable to consider a noise-aware algorithm. On the other hand, the *representational bias* of an algorithm may hinder its applicability for elicitating the process underlying in a log.

Even in the ideal case where the more suitable control-flow discovery algorithm is used for tackling the discovery task, it may be the case that the default algorithm's parameters (designed to perform well over different scenarios) are not appropriate for the log at hand. In that case, the user is left alone in the task of configuring the best parameter values, a task which requires a knowledge of both the algorithm and the log at hand.

In this paper we present a method to automatically assess the impact of parameters of control-flow discovery algorithms. In our approach, we use an efficient technique from the discipline of sensitivity analysis for exploring the parameter search space. In the next section, we charaterize this sensitivity analysis technique

and relate it with other work in the literature for similar purposes done in other areas.

We consider three direct applications of the method presented in this paper:

(A) As an aid to users of control-flow discovery algorithms: given a log, an algorithm and a particular quality metric the user is interested in, a method like the one presented in this paper will indicate the parameters to consider. Then the user will be able to influence (by assigning meaningful values to these parameters) the discovery experiment.
(B) As an aid for recommending control-flow discovery algorithms: current recommendation systems for control-flow process discovery (e.g., [9]) do not consider the parameters of the algorithms. Using the methodology of this paper, one may determine classes of parameters whose impact refer to the same quality metric, and those can be offered as modes of the same algorithm tailored to specific metrics. Hence, the recommendation task (i.e., the selection of a discovery algorithm) may then be guided towards a better use of a control-flow technique.
(C) As a new form of assessing the representational bias of an algorithm: given a log and an algorithm, it may well be the case that the impact of most of the algorithm's parameters is negligible. In that case, then if additionally the result obtained is not satisfactory, one may conclude that this is not the right algorithm for the log at hand.

The rest of the paper is organized as follows: Section 2 illustrates the contribution and provides related work. Section 3 provides the necessary background and main definitions. Then, Section 4 presents the main methodology of this paper, while Section 5 provides a general discussion on its complexity. Finally, Section 6 concludes the paper.

2 Related Work and Contribution

The selection of parameters for executing control-flow algorithms is usually a challenging issue. The uncertainty of the inputs, the lack of information about parameters, the diversity of outputs (i.e., the different process model types), and the difficulty of choosing a comprehensive quality measurement for assessing the output of a control-flow algorithm make the selection of parameters a difficult task.

The *parameter optimization* is one of the most effective approaches for parameter selection. In this approach, the parameter space is searched in order to find the best parameters setting with respect to a specific quality measure. Besides the aforementioned challenges, the main challenge of this approach is to select a robust strategy to search the parameter space. Grid (or exhaustive) search, random search [2], gradient descent based search [1] and evolutionary computation [7] are typical strategies, which have proven to be effective in optimization problems, but they are usually computationally costly. [16,6,3] are examples of parameter optimization applications on a control-flow algorithm. Besides the fact that only a

single control-flow algorithm is considered, all of these approaches rely on quality measurements that are especially designed to work on a specific type of process model.

A different approach, which may also be used to facilitate the parameter optimization, is known as *sensibility analysis* [11] and consists of assessing the influence of the inputs of a mathematical model (or system) on the model's output. This information may help on understanding the relationship between the inputs and the output of the model, or identifying redundant inputs in specific contexts. Sensibility methods range from variance-based methods to screening techniques [11]. One of the advantages of screening is that it requires a relatively low number of evaluations when compared to other approaches. The *Elementary Effect* (EE) method [8,4,5] is a screening technique for sensibility analysis that can be applied to identify non-influential parameters of computationally costly algorithms. In this paper, the EE method is applied to assess the impact of the parameters of control-flow algorithms.

3 Preliminaries

This section contains the main definitions used in this paper.

3.1 Event Log and Process Model

Process data describe the execution of the different process events of a business process over time. An *event log* organizes process data as a set of process instances, where a process instance represents a sequence of events describing the execution of activities (or tasks).

Definition 1 (Event Log). *Let T be a set of events, T^* the set of all sequences (i.e., process instances) that are composed of zero or more events of T, and $\delta \in T^*$ a process instance. An event log L is a set of process instances, i.e., $L \in \mathcal{P}(T^*)$.*[1]

A *process model* is an activity-centric model that describes the business process in terms of activities and their dependency relations. Petri nets, Causal nets, BPMN, and EPCs are examples of notations for modeling these models. For an overview of process notations see [13]. A process model can be seen as an abstraction of how work is done in a specific business. A process model can be discovered from process data by applying some control-flow algorithm.

3.2 Control-Flow Algorithm

A control-flow algorithm is a process discovery technique that can be used for translating the process behavior described in an event log into a process model. These algorithms may be driven by different discovery strategies and provide different functionalities. Also, the execution of a control-flow algorithm may be constrained (controlled) by some parameters.

[1] $\mathcal{P}(X)$ denotes the powerset of some set X.