Designing scientific workflows following a structure and provenance-aware strategy

Abstract

Scientific workflow management systems, (e.g., Taverna, Kepler, Chimera, Galaxy, and Wings) are increasingly being used by scientists to construct and execute complex scientific analyses. Such analyses are typically data-centric and involve "gluing" together data retrieval, computation, and visualization components into a single executable analysis pipeline. Such a pipeline is represented by a workflow which is modeled as a graph, where edges denote scheduling dependencies between computation tasks. Intuitively, a workflow specification is a framework for the execution of workflows, which specifies the set of tasks that are performed and the order to be observed between the different tasks executions. According to the input data given to the workflow specification and assignments of values to the task parameters, different workflow runs are obtained. A run is then also represented as a graph where each vertex represents the execution of a task and edges are labeled by the data consumed and produced at each step. In this thesis, following what is in several workflow systems, we consider that the specifications have a directed cyclic graph (DAG) structure and the runs have the same structures as their specifications. The main goal of scientific workflows is to represent in-silico experiments, which entails frequent reuse and repurposing throughout their life-cycle.

Figure 1 provides (a) an example of workflow specification from Taverna, (b) its representation as a graph and (c) an example of run.

Faced with the increasing complexity of runs and the need for reproducibility

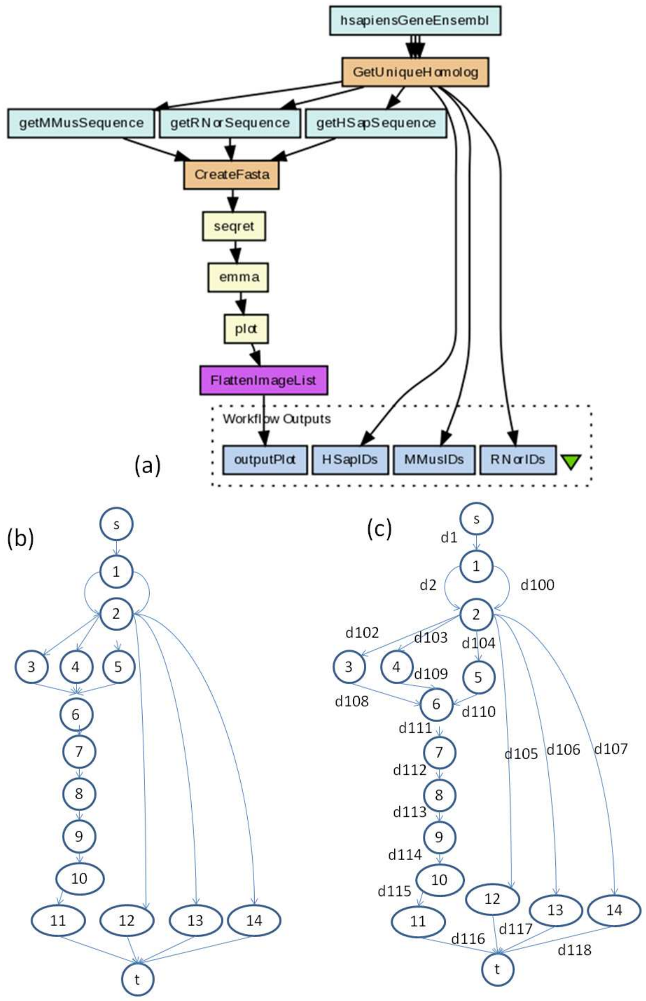

Figure 1: Exemple de workflow issu du système Taverna; (b) représentation sous-forme de graphe; (c) exemple de graphe d'exécution

of results, provenance has become an important research topic. The provenance (also referred to as lineage, and pedigree) of a data product contains information about the process and data used to derive the product. It is often organized as dependency graphs. The visualization of such dependency graphs is especially useful for scientific workflow reuse, since the data, processes, and dependencies associated with a workflow run can be clearly seen by workflow users. By analyzing and creating insightful visualizations of provenance data, scientists can debug their tasks and obtain a better understanding of their results. With the help of provenance, scientists who wish to perform new analyses should be able to find workflow specifications with same or similar meanings of interest to reuse or modify. They can also search for executions associated with a specification to understand the meaning of the workflow, or to correct/debug an erroneous specification. Furthermore, structural provenance queries can help scientists to determine what produced data might have been affected by its input, or to understand how and why the process that led to create a given data has actually failed. Therefore, provenance information is clearly useful for scientific workflows users and systems. However, due to the complexity of workflows, the provenance information which is organized into a graph becomes very large, for which understanding and exploring provenance information becomes a significant challenge for users. While most systems record and store data and process dependencies, a few provide easy-to-use and efficient approaches for accessing provenance information. Additionally, some workflow systems take complex data structure (e.g., lists, trees $\cdots$) into account, which makes provenance presentation a very challenging point. However, to support better reuse of scientific workflows, provenance should be more exploited to present the meaning of scientific workflows for both workflow systems and users.

In the last decade, considerable effort has been put into the improvement of sharing and reusing scientific workflows. Workflow reuse in e-Science is intrinsically linked to a desire that workflows be shared and reused by the community as (part of) best practice scientific protocols. It has the potential to: reduce workflow authoring time (less "re-inventing the wheel"); improve quality through shared workflow development (leveraging the expertise of previous users); and improve experimental provenance at the process level through reuse of established and validated workflows (analogous to using proven algorithms or practices rather than inventing a new which is potentially error-prone). However, as stated by recent studies, while the number of available scientific workflows is increasing along with their popularity, workflows are not (re)used and shared as much as they could be. Several years ago, Goderis et al. summarized several bottlenecks of workflow reuse and repurposing, in which they argue that the main reasons are the restrictions on service availability, lack of a comprehensive discovery model and the complexity of workflows. Recently, Zhao et al. argue that one of the main impediments to workflow reuse is due to the decayed or reduced ability of the resources required for executing workflow, like services and data, which can be either local and hosted along with the workflow or remote, such as public repositories or web services hosted by third parties. The causes of this impediments include: (1) it is difficult to volatile third-party resources; (2) missing example data (it is not always obvious which data can be used as inputs to the workflow execution, and example inputs are often most helpful); (3) missing execution environment (the execution of a workflow may rely on a particular local execution environment, e.g., a local R server or a specific version of workflow execution software); (4) insufficient descriptions about workflows (sometimes a workflow workbench cannot provide sufficient information about what caused the failure of a workflow run). Several solutions for these causes have been provided by Zhao et al.

In this thesis, we have focused specifically on the Taverna workflow management system, which for the past ten years, has been popular within the bioinformatics community. Despite the fact that hundreds of Taverna workflows have been available for years through the myExperiment public workflow repository (http://www.myexperiment.org), their reuse by scientists other than the original author is generally limited. Recently, several studies highlight the complexity of workflow structures as one of the main reason of the limited reuse of (Taverna) scientific workflows. The complexity of workflow structure involves the number of nodes and links but is also related to intricate workflow structure features. Again, several factors may explain such a structural complexity including the fact that the bioinformatics process to be implemented is intrinsically complex, or the workflow system may not provide appropriate expressivity, forcing users to design arbitrary complex workflows. Therefore, to obtain a simpler workflow structure for a complex workflow while preserving the meaning (provenance/semantics) becomes especially important.

The global aim of this thesis is to enhance workflow reuse by providing strategies to reduce the complexity of workflow structures while preserving provenance. Two strategies are introduced.

First, we propose an approach to rewrite the graph structure of any scientific workflow (classically represented as a directed acyclic graph (DAG)) into a simpler structure, namely, a series-parallel (SP) structure while preserving provenance. SP-graphs are simple and layered, making the main phases of workflow easier to distinguish. Additionally, from a more formal point of view, polynomial-time algorithms for performing complex graph-based operations (e.g., comparing workflows, which is directly related to the problem of subgraph homomorphism) can be designed when workflows have SP-structures while such operations are related to an NP-hard problem for DAG structures without any restriction on their structures. The SPFlow rewriting and provenance-preserving algorithm and its associated tool are thus introduced.

Second, we provide a methodology together with a technique able to reduce the redundancy present in workflows (by removing unnecessary occurrences of tasks). More precisely, we detect "anti-patterns", a term broadly used in program design to indicate the use of idiomatic forms that lead to over-complicated design, and which should therefore be avoided. We thus provide the DistillFlow algorithm able to transform a workflow into a distilled semantically-equivalent workflow, which is free or partly free of anti-patterns and has a more concise and simpler structure.

The two main approaches of this thesis (namely, SPFlow and DistillFlow) are based on a provenance model that we have introduced to represent the provenance structure of the workflow executions. The notion of provenance-equivalence which determines whether two workflows have the same meaning is also at the center of our work. Our solutions have been systematically tested on large collections of real workflows, especially from the Taverna system.

In detail, this thesis is organized as follows:

- Chapter 1 gives the introduction of this thesis by stating the motivation, research problems and contributions.

- Chapter 2 presents a collection of mathematical notations used throughout the rest of this dissertation. Based on such notations, the workflow model used in this dissertation is introduced. We then give an introduction to series-parallel graphs and their properties. At the end of chapter 2, we provide an introduction to the workflows of Taverna system, which is the system that we have chosen to mainly work on.

- Chapter 3 starts with related work on provenance models, and then proposes a model to represent the provenance of workflow executions. Later we give a definition of the notion of provenance-equivalence which can be used to identify whether two workflows have the same meaning. Finally, a discussion about extending our provenance model to better support lists of data is given.

- Chapter 4 first gives an in-depth explanation of the motivation of rewriting non-SP workflows into SP workflows. Then we introduce the concept of measuring the distance from non-SP to SP, which inspires some transformation techniques of rewriting non-SP graphs into SP graphs. We then analyze the existing strategies to identify whether they are provenance-preserving and propose a new provenance-equivalent strategy. After that, we introduce the SPFlow algorithm for transforming non-SP graphs into SP graphs and discuss the complexity and soundness of the algorithm. We then demonstrate the feasibility of our approach on real scientific workflows. We finally present a tool with the same name of our algorithm, which takes in a non SP Taverna workflow and provide an SP version of the workflow usable in Taverna.

- Chapter 5 first gives a deep explanation of the second research problem we have considered by presenting several use cases. Then we introduce the antipatterns we have identified and the transformations we propose to do while ensuring that the semantics of the workflow remains unchanged. We then introduce the DistillFlow refactoring algorithm. After that, we provide the results obtained by our approach on a large set of real workflows. Finally, we discuss several points related to our approach.

- Chapter 6 gives the conclusions and the future works.

- Appendix A presents the implementation of DistillFlow and appendix B provides a preliminary study of why scientific workflows have non-SP structures.

As we provide two strategies to rewrite complex scientific workflow structures into simpler ones to make them easier to (re)use, there are two series of contributions which are summarized as below.

First series of contributions (have been published in the 8th IEEE International Conference on eScience 2012 and the 28th Journees de Bases de Donnees Avancees (BDA) 2012):

- We propose a model to represent scientific workflows and provenance generated in their execution.
- We give a definition of the notion of provenance-equivalence which can be used to identify whether two workflows have the same meaning.
- We review several rewriting strategies for transforming non-SP graphs into SP graphs and prove that they are not provenance-equivalent.
- We design a provenance-equivalent algorithm, named "SPFlow", to translate non-SP workflows into SP workflows.
- We illustrate our algorithm by providing an evaluation of our approach on a thousand of scientific workflows.
- We develop a tool based on SPFlow, which takes in a non-SP Taverna workflow and provide an SP version of the workflow usable in Taverna.

Second series of contributions (have been published in the "BMC Bioinformatics" Journal and the 12th International Workshop on Network Tools and Applications in Biology, Nettab 2012 (poster)):

- We identify and automatically detect a set of anti-patterns that contribute to the structural workflow complexity.
- We design a series of refactoring transformations to replace each anti-pattern by a new semantically-equivalent pattern with less redundancy and simplified structure.
- We introduce a distilling algorithm that takes in a workflow and produces a distilled semantically-equivalent workflow.
- We provide an implementation of our refactoring approach that we evaluate on both the public Taverna workflows and on a private collection of workflows from the BioVel project.
- We develop a tool based on DistillFlow, which takes in a Taverna workflow and provide a new version of the workflow which is free or partly free of anti-patterns.

Our approaches (SPFlow and DistillFlow) are currently available for use at https://www.lri.fr/∼chenj/.

Keywords: scientific workflows, provenance, provenance-equivalence, graph rewriting, series-parallel graphs, Taverna, anti-patterns