**Responses to Comments from Reviewer 2**

**NOTE:** The following comments are responded in the revised manuscript using **Blue** coloured text and the same are copied here.

**Comments and Suggestions for Authors:**

This manuscript argues for and presents a model for control flow in scientific workflows. The manuscript reviews both existing scientific workflow models and business process workflow models, and details a description of control flow patterns in scientific workflows. Based on the review, the manuscript identifies requirements that are supported by at least one of three scientific workflow systems (Taverna, Kepler, and Triana), but are not fully specified in the current workflow models ProvONE, OPMW, and Wfdesc. The manuscript proposes the Scientific Workflow Control-flow (SWCf) model to better specify these control flow constructs as an extension of the PROV model. It shows how the constructs in the workflow systems can be handled by the model, and details how the models fare against a use case from agriculture.

Overall, the manuscript does a nice job describing the model and showing how the model expands control-flow handling in existing models. I think the manuscript would be improved by better presenting the problems without this control flow model, by comparing with literature on other workflow systems that handle control flow (e.g. Pegasus) and by making the distinction between the workflow specification and the execution clearer.

== Comments ==

***Response:*** We are thankful to the reviewer for the valuable suggestions and recommendation. We acknowledge that the reviewer’s suggestions have improve the overall quality of this article. We have responded to the suggestions in the following way:

***Comment 1:* \*** The progression from workflow patterns to requirements to an examination of how this works in existing models and systems is very nice, and I think this is of significant value.

***Response:*** Thanks for the appreciation.

***Comment 2.1:*** \* The motivation for the manuscript seems to be that control flow workflows cannot be properly modeled, but the introduction does not cite real-world examples.

***Response:*** We agree to the reviewer; however, to keep the introduction section concise and engaging we have provided the limitation of the existing provenance models in the introduction section (line 76-91 on Page 2) and a complete example is provided in Section 5.3.

***Comment 2.2:*** Similarly, the argument that "underspecified" workflows raise problems is not well-established. The manuscript concedes that for retrospective provenance is unaffected, but the first stated problem is that an experiment is misinterpreted. It seems that with retrospective provenance, it would still be clear.

***Response***: Thanks for the comment. We have made it clear in the paper that **the design of** an experiment is misinterpreted not the experiment itself. (see line 98-100 on Page 3)

* “Firstly, the publication of an underspecified workflow structure results in a misinterpretation of the design of a scientific experiment (i.e., workflow design), which hinders the reusability of the workflow and its results.”

***Comment 2.3:***

The third argument is that it is impractical to validate output seems unfounded to me. Why does the control-flow structure make it impractical to validate output? This seems completely orthogonal to whether the control flow is totally specified.

***Response:*** The following text is changed in the revised manuscript to make our point clear for the reader (see line 103-104 on Page 3)

* “Finally, the reproducibility of a scientific workflow to independently confirm its output using new data is hard without the accurate and complete structure of the workflow.”

***Comment 3:*\*** The Pegasus work by Deelman et al. ("Pegasus, a Workflow Management System for Science Automation") seems like an important reference point. Here, scientists can encode dependencies but the system can translate an abstract workflow to a concrete implementation that can be scheduled on the cloud. The control flow (which jobs are run in parallel, etc.) may be dependent on resources so the retrospective provenance may be more critical than the prospective provenance.

***Response:*** We have included the following text in the revised paper to mention this work (see line 88-91 on Page 3) and (see line 133-136 on Page 4).

* “Moreover, Pegasus translates user-defined abstract workflows into a concrete implementation required to schedule the workflow on the cloud by introducing some control constructs, like parallel execution.”
* “Pegasus is another popular SWfMS, but it involves control-constructs only as a part of the execution plan, which depends on the availability of resources. A user-defined workflow and its prospective provenance do not include control constructs semantics; thus, we did not consider control-flow patterns supported by Pegasus.”

***Comment 4:*** \* Along with this, the manuscript should make a clear distinction between prospective and retrospective provenance and better describe how SWCf relates to retrospective provenance. It seems like it may be no different than without the model?

***Response:*** Thefollowing text is added in the revised version of the paper to define prospective and retrospective provenance (see line 39-47 on Page 2).

* “Workflow provenance is distinguished as prospective provenance and retrospective provenance. Prospective provenance captures the structure and static context of a workflow. It is independent of workflow execution or input data and provides an abstract overview of a workflow. Prospective provenance solutions should reveal both the control flow as well as dataflow dependencies. Retrospective provenance captures the information about a workflow's execution that becomes available when running the workflow. This information includes facts about the execution of each workflow step as well as the runtime environment. These two forms are independent of each other; hence, a provenance solution may support only one or both of them.”

The model proposed in this article is aimed for capturing workflow specification patterns only. We argue that the missing control constructs semantics affect only prospective provenance and our model can capture workflow specification patterns only. However, to capture the execution provenance, the proposed model can be integrated with the existing provenance models e.g., ProvOne, Triana, Triverna. The following text is included in the paper to make understandings clearer (see line 92-97 and line 105-109 on Page 3) .

* “The retrospective provenance of a control-flow workflow does not include workflow control constructs as the execution trace itself typically specifies the activity and the discrete-time points at which the activity has happened. Therefore, overlooking control constructs in existing provenance models do not affect their ability to capture the retrospective provenance (i.e., the information about a workflow execution). However, it results in underspecified workflows (i.e., incomplete prospective provenance) that raises several problems”
* “To address these problems, we identified control-flow patterns that a model must be able to describe to be an acceptable model for scientific workflows prospective provenance. Further, we present a provenance model to capture prospective provenance for scientific workflows that can specify control-flow patterns and provide the means to integrate this model with existing provenance models.”

***Comment 5.1:* \*** In Section 4, the reasons for divergent results would seem best informed by retrospective (and prospective) provenance.

***Response:*** The following text is added in the paper which highlight that both types of provenance are required for most of the analytical queries (see line 422-423 on Page 12).

* “Workflow prospective and retrospective provenance are required to address many provenance analytic queries”

***Comment 5.2:*** Are there cases where the retrospective provenance looks the same but the prospective provenance is different?

This is another analytical query. We have included the following text in the revised submission to address such a situation (see line 429-434 on Page 12).

* **“Or is the prospective provenance different for two similar retrospective provenance traces of a workflow?** To answer these provenance queries, the provenance management component of SWfMSs must capture a correct and complete workflow specification provenance along with the execution provenance. The proposed model has the ability to capture a complete and correct workflow prospective provenance as discussed in Section 5.”

***Comment 6:* \*** It seems worthwhile to discuss which requirements can be translated to dataflows. For example, R1, R2, and R3 can be implied by (dummy) data output links between the modules. Certainly, some of them cannot, and that is the value in this model.

***Response:*** We have included thefollowing text in this revision which defines prospective and retrospective provenance (see line 581-590 on Page 20).

* “These provenance descriptions show that the state-of-the-art scientific workflows provenance models are unable to correctly and completely specify all control-flow driven scientific workflows. Since sequence, parallel split, and synchronisation patterns describe the unconditional flow of data ‘to’ and ‘from’ task(s); hence, these patterns can be considered as data-flow patterns. A provenance model designed for the data-driven scientific workflows can model these patterns as data-flow patterns, as shown in Figure 7. However, all other patterns (i.e., R4,…, R11) involve a condition on the follow of data. The data-driven scientific workflows provenance models are unable to capture prospective provenance for these patterns because they lack a class or/and a property required to model the conditional constructs like if/else or case structure.”

***Comment 7:*\*** The use case in Section 5 may be stronger if the control flows were more complex. The conditional is nice, but the four parallel computations can be modeled by a dataflow without issue. Similarly, it's not clear the display loop is necessary (i.e. would it actually be implemented like that).

***Response:*** We provide a real-world in-use workflow example and its provenance descriptions. We choose this workflow because it involves more than two control-flow patterns and its provenance descriptions could have been comprehendible to a reader. The provenance descriptions for a more complex workflow is both difficult to present in paper and understandable for a reader. However, we have provided the provenance descriptions for all the considered patterns in Section 5.1(see line 447- 485 on Page 13).

***Comment 8:*\*** The manuscript lacks a discussion of the limitations of the model. To me, Figure 7d shows more complexity that is harder for end-users to interpret. Perhaps other graphs (with abstraction) would make this easier?

***Response:*** The following text is included in the revised paper to address this comment (see line 598-614 on Page 20)

* “We showcase the correctness of the proposed model using a real-world agriculture workflow. Our model successfully captured the control constructs semantics in the provenance description of the workflow which the baseline models are unable to capture. However, a significant concern is that considering control flow constructs in scientific workflow provenance model increases the complexity of the model; it encodes a complete control flow modelling language in the model. This complexity poses the following two challenges: an increase in the volume of provenance data required to capture the prospective provenance of a workflow, and a difficulty in interpreting the provenance descriptions for end-users.

Table 3 provides some insights on the size of the provenance descriptions for the example GHG workflow based on both the baseline and SWCf provenance models. The statistics show that SWCf aligned provenance description doubles the amount of data as compared to ProvONE and OPMW, but the Wfdesc aligned provenance description size is almost the same to our model. Also, with the advent of more sophisticated and scalable triple-stores and graph databases, handling a few extra triples capturing the control flow semantics in scientific workflows is practically not a problem. In addition, provenance descriptions are processed and interpreted by machines and a more readable form or visualisation can be generated for a user-friendly interpretation of these descriptions.”

***Comment 9:*\*** I feel like the paper would benefit from a more concise description of the major components shown in Figure 2. The inline Manchester Syntax in Section 3 would be better moved to an appendix or code listing figures.

***Response:* We** appreciate reviewer’s suggestion. We moved the syntax to code listings as suggested to improve the clarity of the proposed model.

***Comment 10:* \*** swcf:Controller feels more like a prov:actor to me, and it would be nice to have a justification for making it a prov:entity.

***Response:*** The following text is included in the revised submission to address this comment (see line 308-310 on Page 9)

* “The swcf:Controller class represents a concept for specifying the sequence of connected tasks instead of being responsible for an activity; therefore, we consider it a sub class of prov:Entity rather a prov:Agent (i.e., swcf:Controller ⊂ prov:Entity).”

***Comment 11:***

\* In a number of locations, "SWCf model" should be "the SWCf model".

\* p. 6: "worth-noticing" -> remove dash

\* p. 6: "othercontrol" -> add space

\* p. 8: "later specifying" -> "latter specifies"

\* p. 8 "specifies the execution of" -> "specifies that the execution of"

\* p. 9: "and itself a controller" -> "and is itself..."

\* p. 14: "The most of SWfMSs" -> "Most SWfMSs"

***Response:*** Thank you, all the suggestions has been incorporated.