

# Reviewer(s)' Comments to Author

January 2, 2019

## 1 Reviewer: 1

### 1.1 Comments to the Author

This is timely and a really nice piece of work. This Cis-interface has the potential to become a major tool to support the future development of the plant systems model integration. I have a few specific comments on this and would like the author to address.

- ☐ First, it would be great to have a more clear description of the novelty of the pipeline developed. Is it mainly a compilation of existing tools, in that case, this needs to be clearly indicated. If not, then a clear statement in this might be great.
- ☒ Secondly, the work will benefit a lot from a more explicit description of a user case. For example, clearly write out the exact commands needed to finish one round of code integration.
- ☒ Thirdly, I would be keen to show some performance data of integration of different modules. No data are provided on this at this stage.
- ☐ Fourthly, the main purpose of this paper is to describe a platform which can help plant modelers of different research areas and using programming languages work together more effectively and easily, it would be great if author can explain the advantage of this new software from the perspective of users.
- ☐ Fifthly, many of the software engineering methodologies described in the paper are widely used in the field of software engineering. It would be better to provide some more concrete description of how this particular tool help users.
- ☐ Finally, it would be helpful to provide a flow chart and pseudo-code of the core algorithms used to link different modules.

## 1.2 Response from the Author

We would like to thank the reviewer for their comments as we believe they helped make the paper much stronger and to better represent the work being described. We will address the reviewer’s comments in the order they were raised:

1. ?
2. We whole heartedly agree with the reviewer that the manuscript greatly benefits from a concrete use case. As such, we have added a new section “Worked Example” after the methods section that walks through, step-by-step, integration of the example models presented in the introduction. The code from the example is available on the paper branch of the `cis_interface` GitHub repository and will be published as part of the documentation during the 1.0 release.
3. We believe that the reviewer is referring to performance data on integration of specific biological *models*, but would appreciate some clarification if this is not the case. While we would like to show concrete examples of the performance of some integrations done with `cis_interface` using actual models, most of the models that `cis_interface` has been used with are still in the pre-publication stage. As the code for the models is not yet public, these performance tests would not be informative to those without access. In addition, the performance increase due to parallelism is highly dependent on how coupled the models are and how many models are part of the integration. If two models are integrated as in the case of the added root/shoot integration example as they are, the shoot model will always be waiting for the root model to complete its calculation and there not be a speedup over scripted integration of the two models via an intermediary file. However, if the shoot model had other calculations to perform before it needed the data from the root model, a speedup would be possible and completely dependent on the ratio of independent calculations to dependent calculations and the time required for model communication. The speedup calculation is even more complex when more than two models are involved. To reflect this complex relationship, we have added a subsection to the end of the results section entitled “Speedup from Parallels” that discusses this and present an equation for calculating the speedup for an integration of two models given the appropriate parameters.
4. ?
5. ?
6. ?

## 2 Reviewer: 2

### 2.1 Comments to the Author

This is a very well written manuscript that is accessible to people from different backgrounds. Few minor comments:

- ✓ Background: Pg 4, L8-18: The first paragraph contains broad, bold statements with no citations as examples. To move away from a largely anecdotal paragraph, examples from the literature should be given. It would also help to cite the Crops in silico perspective article in Frontiers (<https://www.frontiersin.org/articles/10.3389/fpls.2017.00786/full>).
- ✓ P4, L35-38: give examples here too. What solutions?
- ✓ P6 line 17: a to "as"
- ✓ P7 line 9: remove extra "tools"
- ✓ In the conclusions, it would be helpful to have a few lines suggesting uses or applications of the Cis interface. For example, researchers could use `cis_interface` as a user-friendly way to run model code either from the published literature or when reviewing manuscripts. Likewise, experimentalists can update parameter files based on in-house experimental data and then run the models, even if they have no prior expertise in modeling.

### 2.2 Response from the Author

We would like to thank the reviewer for their comments as we the manuscript is stronger as a result. We have responded to the reviewer's comments in the order they were posed.

1. We agree with the reviewer that the paragraph in question was vague and have added several references to examples from the literature and including the Crops in silico perspective article.
2. The sentence in question "Within both computational biology and the larger scientific computing community, groups have developed around different solutions to the problem of connecting models." was the last sentence in the introductory part of the background section and was meant to refer to the solutions described from the literature in the following subsections. We have clarified this by explicitly listing the solutions that are described in those sections.
3. We have added several sentences to the conclusions discussing the use cases for `cis_interface` including as a tool for integrating models and as a tool for executing models without knowledge of the model's programming language. While we agree

with the reviewer that `cis_interface` could be used for peer review and collaboration by exposing models via a standard interface, we believe this application will be most powerful when used with the model integration GUI (which is separate from the `cis_interface` package itself) or by calling models as functions from Jupyter notebooks (which is a feature that was added after submitting the manuscript and will be released in version 1.0 of `cis_interface`). As such, we have added a subsection for "Models as Functions" in the "Current/Future Improvements" section and discuss this application further there and in the "Graphical User Interface" subsection.

### 3 Reviewer: 3

#### 3.1 Comments to the Author

This manuscript reports on efforts to create a universal interface toolbox for integrating computational models across languages and scales. This topic is without doubt one of the key topics in further development and application of existing and to-be-developed plant models. I appreciate the authors' effort in developing an open source Python package `cis_interface`, which can be installed on 3 different operating systems and support integration of models written in 4 different programming languages. I think the paper has great potential but I do have some comments:

- The author had a lengthy discussion on comparison of communication time cost for model integration with different communication mechanisms, programming languages, Python versions and operating systems, actually this is also the main result of current manuscript. However, without comparison with existed tools, I think this part should be greatly condensed. I believe readers will be more interested in how does `cis_interface` work for combining different types of biological models into a complex network. These models may have different time step length (e.g. ODE vs constant step length) and different types and scales of input and output which need complex data conversion (e.g. total leaf nitrogen content to different protein levels). Moreover, how to perform integration of input for one model from different models? For example, canopy model only gives 3D mesh data with light level on each facet, however, metabolic model needs both light level and protein levels (calculated from another model) for each facet, how to correctly integrate the protein data and the light data?
- I expect more detailed biological modeling examples to be provided, from which the readers are able to know how to use the `cis_interface` and what's the advantage of it. I also expect the author to suggest some general modeling principles from following which the newly developed model can easily be integrated with other models.

### **3.1.1 Minor comments**

- ✓ Page 7 line 9: remove redundant ?tools?

### **3.2 Response from the Author**