



EGUsphere, author comment AC3  
<https://doi.org/10.5194/egusphere-2022-802-AC3>, 2022  
© Author(s) 2022. This work is distributed under  
the Creative Commons Attribution 4.0 License.

## Reply on RC2

Thomas Bossy et al.

---

Author comment on "Pathfinder v1.0.1: a Bayesian-inferred simple carbon-climate model to explore climate change scenarios" by Thomas Bossy et al., EGU sphere,  
<https://doi.org/10.5194/egusphere-2022-802-AC3>, 2022

---

Dear Mr Enting,

Thank you for all your comments. Please read here our answers to all of them.

### Comment #2.1:

- GENERAL COMMENT

The editors have advised me that this paper has been submitted to GMD as a 'model description' paper. Accordingly, I have assessed it against the GMD criteria for such papers, (in `about/manuscript\_types.html` on the GMD website)

In the primary text, I find the introduction is short of detail on the context and that the detail about accessing the model is insufficient.

### Response #2.1:

The detail about accessing the model were indeed insufficient, and we have added those in an update of the model's repository.

Regarding detail on the context, however, we are unsure what the referee expects, exactly. We explain the philosophy behind this type of models, the philosophy behind our own model, and cite a number of papers on the topic. We propose to add a paragraph in the Introduction to better position Pathfinder with respect to other SCMs: "*Compared to other SCMs (Nicholls et al., 2020), Pathfinder is much simpler than models like MAGICC (Meinshausen et al., 2011), OSCAR (Gasser et al., 2017) or even HECTOR (Hartin et al., 2015). It is comparable in complexity to FaIR (Smith et al., 2018) or BernSCM (Strassmann and Joos, 2018), although it is closer to the latter as it trades off an explicit representation of non-CO2 species for one of the carbon cycle's main components. This choice was made to help calibration, keep the model invertible, and be compatible with IAMs such as DICE (Nordhaus et al., 2017). While most SCMs are calibrated using procedures that resemble Bayesian inference (Nicholls et al., 2021), Pathfinder relies on an established algorithm whose implementation is fully tractable, and that allows for an annual update as observations of atmospheric CO2 and global temperature become available.*"

**Comment #2.2:**

The specifications note that the text should describe the software and hardware requirements for running the model. This is missing.

**Response #2.2:**

Indeed. We propose to add this information in Appendix:

*"Pathfinder has been developed and run in Python (v3.7.6), preferentially using IPython (v7.19.0). Currently, packages required to run it are NumPy (v1.19.2), SciPy (v1.5.2) and Xarray (v0.16.0), and it has hard-coded dependencies on PyMC3 (v3.8) and Theano (v1.0.4) that are in fact used only for calibration. Other versions of Python or these packages were not tested.*

*The calibration procedure takes about 9 hours to run on a desktop computer (with a base speed of 3.4 GHz). Simple use of the model is much faster: the idealized experiments and SSP scenarios for this description paper, which represent 2984 simulated years, were run in about 20 minutes for all 2000 configurations and on a single core. A single simulated year takes a few tenth of a second, although a number of options in the model can drastically alter this performance. Note also that this scales sub-linearly with the amount of configurations or scenarios because of the internal workings of the Xarray package, albeit at the cost of increased demand in random-access memory."*

We also propose to add in Appendix the changelog and 'known issues' that we added in the updated README file.

**Comment #2.3:**

The specifications say that a publication should consist of text, code and user manual. The authors acknowledge the incomplete nature of the current state of their site, describing it as "messy". I have downloaded and expanded the zip file and extracted files manual.pdf and manual.md. As a user manual, I find it inadequate because it is only a brief download instruction, a list of model variables and a list of equations that presumably repeat those in the text and essentially nothing about how to use it. The latter two sections are faulty (unreadable) as described below. If the authors want people to try the model, they need to do better.

**Response #2.3:**

We are very sorry we did not see this error when exporting the .md files into .pdf. This has been corrected in version 1.0.1 (in both the GitHub and Zenodo repositories). The new Zenodo repository is available here: <https://doi.org/10.5281/zenodo.7003848>

However, this was merely a displaying error, and most of the requirements were already provided as they should have been: main text as a description of the model, and code fully usable. We acknowledge, however, there was room for improvement in making the readme and manual files more user-friendly, which we did in this small update.

**Comment #2.4:**

My recommendation is that the authors withdraw the paper and resubmit when the online

information is better organised. Releasing the model at this stage is doing a disservice to the modelling community and the authors themselves.

#### **Response #2.4:**

We find that greatly exaggerated. The small display error is now corrected, and additional information including a very simple example of use has been added.

#### **Comment #2.5:**

- SPECIFIC COMMENTS

2.1 Without doing a line-by-line check, it seems to me that the description is sufficient to enable someone else to implement equivalent code to integrate the model equations.

The ability to run with multiple choices of two input series is more problematic (without actually writing 6 separate versions (sharing some code) using the principles described by Wigley 1991).

There does not seem to be enough information to independently implement the model calibrations. (This is not necessarily a GMD requirement).

#### **Response #2.5:**

The ability to run with multiple choices of two input series was indeed implemented through separate versions of the model (we actually implemented only 4 of the 6 possibilities). The code is in the files whose names start with 'mod\_' located in the 'core\_fct' folder. We propose to add the following detail to the manuscript: "*(in terms of code, implemented as separate versions of the model)*".

Regarding the calibration, it is simply not true: one can execute the 'run\_calib\_and\_hist.py' file located in the 'run\_scripts' folder to repeat exactly the calibration. One can also backtrack one's way into the various functions and sub-functions called when executing said script, to further pick the calibration procedure apart.

All this is now easier to figure out, thanks to the clean and expanded manual file.

#### **Comment #2.6:**

2.2 The manuals and online information.  
(This is based on the contents of the zip file from the zemodo repository).  
The manual consists of 3 sections:

- \* Run a simulation
- \* Notation
- \* Equations

##### 2.2a. Manual.pdf

Many aspects are not being displayed, even in the list of variables. My guess is that the pdf has been created in a form that makes use of external fonts on the author's system, rather than importing the requisite font descriptions (maybe for math fonts) into the pdf file. The whole of the equation section fails to display.

**Comment #2.6:**

Corrected. Apologies again for this. This was because of failed image imports, as until very recently the GitHub .md interpreter did not support math display.

**Comment #2.7:**

2.2b. Manual. md. This is the other way round. The equation section displays as math characters, presumably reproducing what is in the text. However the Notation section is just an unformatted, unspaced string of ascii characters with occasional math symbols, presumably those missing from column 1 in the equation section in manual.pdf.

**Comment #2.7:**

Corrected.

**Comment #2.8:**

2.3 The online files. (From the .zip)

As noted, the "manual" does not describe what is needed to run the model, in particular what input files are involved? For example what is the role of the .nc (netcdf?) files and the .xlsx files.

How do these relate to the RCMIP standards noted by Nicholls 2020, 2021? Similarly, the comments at the beginning of each python file give no information about what the file is for and even if it is the main file or a module called by one or more main files.

**Comment #2.8:**

Now provided in the expanded manual file, specifically as a list of folders and/or files and a brief description of the content.

We note here that the file formats we use are not related to the RCMIP standards (that we find inconvenient).

**Comment #2.9:**

2.4 Role of model.

The authors state that they have identified the need for a new model, citing Nicholls 2020.

This is not obvious from that paper. More detail should be given, being more specific about the niche that the model is intended to fill and the relation between this paper and earlier work such as Meinshausen 2009.

**Response #2.9:**

In the introduction, we clearly state the three requirements we want our model to fulfil as none of the existing model fulfil them simultaneously (line 21-25): *"Pathfinder was designed to fulfil three key requirements: 1. the capacity to be calibrated using Bayesian inference, 2. the capacity to be coupled with integrated assessment models (IAMs), and 3. the capacity to explore a very large number of climate scenarios to narrow down those*

*compatible with limiting climate impacts.”*

We propose to add a paragraph to better position Pathfinder with respect to other SCMs: *“Compared to other SCMs (Nicholls et al., 2020), Pathfinder is much simpler than models like MAGICC (Meinshausen et al., 2011), OSCAR (Gasser et al., 2017) or even HECTOR (Hartin et al., 2015). It is comparable in complexity to FaIR (Smith et al., 2018) or BernSCM (Strassmann and Joos, 2018), although it is closer to the latter as it trades off an explicit representation of non-CO2 species for one of the carbon cycle’s main components. This choice was made to help calibration, keep the model invertible, and be compatible with IAMs such as DICE (Nordhaus et al., 2017). While most SCMs are calibrated using procedures that resemble Bayesian inference (Nicholls et al., 2021), Pathfinder relies on an established algorithm whose implementation is fully tractable, and that allows for an annual update as observations of atmospheric CO2 and global temperature become available. ”*

**Comment #2.10:**

2.5 Mixed layer pools.

The other referee raised the issue of the meaning and number of the mixed layer pools. This structure has no physical meaning and simply represents a mathematical transformation to go from the impulse response function of Joos 1996 to a set of differential equations. This is presumably a well-known transformation, although I first became aware of it through Wigley 1991. The number of pools derives from the number of exponentials in the impulse response function. An approach (derived from electrical engineering) for deriving lower order approximations is given in my recent paper Enting 2022.

The structure has the 5 pools operating in parallel. Writing the equations in matrix vector form as  $dx/dt = A x + b.f$ , where  $f$  is the input, it should be possible to take linear combinations (applied to both rows and columns of  $A$ ) such that only one element of  $b$  is zero, and maybe even end up with  $A$  in the tridiagonal form representing pools in series.

**Response #2.10:**

We thank the referee for the reference, and will look into it when we work on revamping and hopefully simplifying the ocean carbon module.

We also note that  $A$  is already a diagonal matrix, therefore the pools are solved independently from one another, and we do not see any way to make the calculation faster (except, of course, reducing the number of pools).

**Comment #2.11:**

2.6 Calibration

At one point it is noted that the philosophy that priors are defined by the more complex earth system models, with posterior values determined by applying observations as constraints. If this is true throughout, it should be stated in both the abstract and the introduction. Of course, this sort of Bayesian calibration is vulnerable to the likelihood that such priors will be drawing on some of the same information as the constraint data.

**Response #2.11:**

We propose to add this sentence upfront in the introduction (line 31-32): *“The philosophy*

*of Pathfinder is to use complex models as prior information and only real world observations (or assessment combining many lines of evidence) as constraints for the Bayesian calibration."*

The referee makes a good point regarding degenerate information. So we also propose to add the following brief discussion in the 'principle' subsection on the Bayesian calibration:

*"Such a Bayesian calibration is vulnerable to the possibility that the priors draw on the same information as the constraints. However, given that Pathfinder is a patchwork of emulators whose parameters are obtained independently from one another and following differing experimental setups, we expect that the coherence of information contained within the priors and the constraints is very low. Our choice of using only complex models as prior information and only observations and assessments as constraints also aims at limiting this vulnerability."*

**Comment #2.12:**

An extra sentence on the PyMC3 operation would be helpful.

**Response #2.12:**

We propose to significantly expand this section. In addition to the short paragraph of Response #2.11, comments by Referee #1 motivated us to add the following sentences:

*"Summarily, the Bayesian calibration updates the joint distribution of parameters to make it as compatible with the constraints as possible given their prior estimates, which increases internal coherence of Pathfinder by excluding combination of parameters that are unlikely."*

and

*"The Bayesian procedure itself is implemented using the Python computer language, and specifically the PyMC3 package (Salvatier et al., 2016). The solving of equation 47 and its normalization are done using the package's full-rank Automatic Differentiation Variational Inference (ADVI) algorithm (Kucukelbir et al., 2017), with 100,000 iterations (and default algorithm options). The choice of variational inference instead of Markov chain Monte Carlo is motivated by the significant size our model (Beir et al., 2017) and the speed of ADVI. An additional strength of the full-rank version of the ADVI algorithm is its ability to generate correlated posterior distributions even if the prior ones are uncorrelated. Convergence of the algorithm was controlled through convergence of the ELBO metric (Kucukelbir et al., 2017). All results presented hereafter are obtained through drawing 2000 sets of parameters – that we call configurations – from the posterior or prior distributions."*

In this section on technical implementation of the Bayesian calibration, we also propose to explicit the IMEX solving scheme we use for the differential system.

**Comment #2.13:**

2.7 Input time series.

It is unclear what happens to the two input time series (which would seem to be needed at all times) as the calibration adjusts the 6 parameters that characterise these series.

**Response #2.13:**

What happens to the input time series during calibration is illustrated in the first two panels of Figure 8. In addition, the changes in the distributions of the 6 parameters (between prior and posterior) are shown in Figure 5, in the last 6 panels.

**Comment #2.14:**

3 TECHNICAL CORRECTIONS

Line 296. Cross reference not working.

**Response #2.14:**

Done.

**Comment #2.15:**

Line 440. to be -> from being

**Response #2.15:**

Done.

**Comment #2.16:**

Line 549 stricken -> struck

**Response #2.16:**

Done.

**Comment #2.17:**

Caption for table A1 notes an indicator of prognostic variables. This does not seem to be used.

**Response #2.17:**

Removed.

**Comment #2.18:**

I have not made any attempt to check against the guideline that papers should use either US or UK English, but not mix them.

**Response #2.18:**

This will be checked during copy editing.

Thomas Bossy, 20 October 2022