

Response to Reviewers: CPC initial submission

Note, reviewer comments are included with responses in blue text.

Reviewer 1

This manuscript introduces “RadLib,” a C++ library to calculate absorption coefficients for use in radiative heat transfer calculations. The library has reasonable value to the research community, despite a few limitations in its scope and design. The library is self-contained (doesn’t have any external dependencies), which is convenient.

I have a number of comments on the manuscript as well as the software itself which the authors should consider to improve the manuscript and software.

The authors thank the reviewer for their time and careful review.

Manuscript:

1. There is no general user documentation provided in section 3. Figure 1 includes information that is more algorithmic than illustrative of the API.

Section 3 has been significantly revised and extended to include more information on how users can include and interact with the RadLib library. This includes a description of the build procedure and a list of installed files. Compilation of user code with RadLib is outlined. A description of the interface functions is also provided. The key interface function was `get_k_a(...)`. An additional function is now included and described, `get_k_a_oneband(...)`. The function arguments are now described. Since the initial submission, a Fortran 90 interface has been added and this is included in the section.

2. Section 5 makes several editorial claims that are quite subjective, and not well-supported:
 - i. First sentence: “historically difficult to implement ... complexity and high computational costs”
 - o Property evaluation is not the expensive part of a radiation calculation. Solving the RTE is the expensive part.
 - I wouldn’t characterize the property evaluation as particularly complex in comparison with other aspects of combustion simulations (kinetics, thermodynamics, transport properties).
 - ii. At the end of the first paragraph, the authors seem to imply that RadLib will solve the challenges of radiation modeling. Again, it is the RTE solver, not the property evaluation, that is the largest time-sink in terms of computer and human resources.

- iii. These kinds of statements persist throughout the second paragraph as well. There are limitations in RadLib such as lack of scattering treatment for particle-laden flows, inability to include more species, lack of any line-by-line capabilities, etc. The library checks a few useful boxes, but the verbiage here seems to take more credit than the authors may be due.

In the first sentence and paragraph of Section 5, “radiative heat transfer models” does not refer specifically to either radiation property models or RTE evaluation methods, but the treatment of computational radiative heat transfer as a whole. This generalization was intended for the entire first paragraph of Section 5.

Regarding solution of the RTE versus the radiative property model, the reviewer is correct that in most engineering calculations, specification and implementation of the RTE can be more complex and costly than that of the property model. For both parts, however, this depends strongly on the particular model used, (and the geometry considered). The optically-thin model RTE model is trivial, and the RCSLW property model is not. We revised the manuscript to include a discussion of this in the second paragraph of the introduction, with a specific note that “In practical engineering simulations, the solution of the RTE is often the most complex and costly part of radiative simulation...” That said, we believe that access to good radiative property models as presented in this paper is a valuable contribution.

We want to be careful with what is claimed in the paper, and some major revision has occurred in response to this reviewer’s comments and others. Section 4 contained examples provided along with results, but now contains a description of the examples provided with the code. Section 5 contained some discussion items as noted by the reviewer, but now includes simulation results for the examples along with a CFD demonstration. The discussion items noted by the reviewer have been modified and folded into the Conclusions section.

3. Section 5 claims that “RadLib’s modular framework is designed to easily accommodate new models as well,” which is a bit of an overstatement. See comments below w.r.t. assumptions on species considered, hard-coded polynomial coefficients, etc. all of which reduce extensibility.

We remove the word “modular” throughout the paper. We do believe that the library can be extended, and have noted this, with softer language, in the revision. The reviewer is correct that we do not provide explicit functionality, for, e.g., generic extension to other species. In some cases, like the PM model, such extension would be trivial within the existing code structure. In the RCSLW model, extension to other species would require provision of the species ALBDF (not trivial), and then extension would be a straightforward, but the code is not set up for arbitrary species in arbitrary order. The library is primarily targeted at combustion applications where CO₂ and H₂O are the primary radiative species. Future efforts may motivate further extension.

4. There are several minor issues that need to be addressed:

- i. Abstract: “well-written” is subjective and should be removed.

Done.

- ii. Introduction: second sentence: “Combustion simulations...” is a run-on sentence.

We have revised the wording in the introduction.

- iii. Page 10: “[INSERT RUNTIME COMPARISONS HERE]”

This has been corrected.

- iv. Section 7 heading: “Interes”

This has been corrected.

Software deficiencies & suggestions:

- 1. API doxygen documentation is non-standard. It appears that the doxygen documentation is provided in the .cc files rather than the header files, which is where it is typically found. Downstream developers would typically be working from headers and the installed library, so it is convention to have doxygen documentation in the header only. This also reduces clutter in the implementation files.

We believe this to be a stylistic preference. Here’s a quote from the *doxygen website*: “Unlike most other documentation systems, doxygen also allows you to put the documentation of members (including global functions) in front of the definition. This way the documentation can be placed in the source file instead of the header file. This keeps the header file compact, and allows the implementer of the members more direct access to the documentation.” Furthermore, Doxygen is primarily used to build external documentation (html), so, from that perspective, it does not seem as important where the Doxygen comments are in the source code. The reviewer’s argument makes more sense for libraries that are distributed without the source. In our case, we provide the source and the header files. We follow the approach of putting class comments, data member documentation, and class usage in the header, and documentation of how the methods/functions work in the source files where they are defined. This has the benefit of not cluttering the header files. There certainly is some precedence for this in online discussions.

- 2. `make_examples.sh`:
 - i. will only work with a g++ compiler in the user’s path.
 - ii. Is referred to as `build_examples.sh` in the [README.md](#) file
 - iii. is redundant with the cmake build system and isn’t portable. It should probably be removed.

The `make_examples.sh` file has been removed and examples are built with CMake. This script was originally included for reference.

3. It appears that there are no tests associated with the library. This is a bit surprising - I would expect regression test coverage on the basic API functionality.

The API is pretty simple and consists of two functions. Extensive examples are provided, which are compared to LBL data, and which, as published here, are identical to the results of the published radiation models. That said, a branch of the code is being developed that includes tests.

4. CMake build system:

- i. Consider installing a “RadLib.cmake” file for downstream usage by CMake-based projects. This helps downstream build systems configure for RadLib usage (setting include paths, etc.).

This is a good point. The installation now includes `radlib.cmake` files, located by default in the `installed/cmake/` directory, for downstream usage.

- ii. probably shouldn’t specify optimize or debug flags directly, as CMake will provide appropriate values on most platforms.

The CMake build has been updated to reflect this.

- iii. the default build type isn’t defaulting to Release for me; it remains blank.

This appears to be working in the revised version of the code. Testing shows the value of `CMAKE_BUILD_TYPE` as set to Release if not otherwise set. This is specified at the very top of the base `CMakeLists.txt` file. Note that the build type can be overridden by IDE build settings, among other things, and that this may be a system issue specific to the reviewer.

- iv. I suggest not using verbose makefiles by default.

The build has been updated so that verbose makefiles are no longer the default.

- v. Using `GLOB` to install the headers is not the appropriate CMake approach.

The build updated accordingly; `GLOB` is no longer used to install header files.

- vi. the data files are not copied into the examples build directory, meaning that the example executables do not run.

The original code has been updated and tested and this is now corrected.

- vii. The `run-examples.sh` script is not helpful as it is not installed into the build directory where the executables are produced and doesn’t have path information to actually run the executables.

Executables for the examples are installed within the example folders (in addition to the build directories), so `run_examples.sh` runs without any issues.

- viii. Using “doxygen” as a target is not ideal since it conflicts with the “doxygen” executable name. It still works fine though...

Build target name has been changed from “doxygen” to “docs” for safety. Note that building the documentation is off by default.

- 5. It appears that line-by-line data from HITEMP/HITRAN files cannot be loaded into RadLib. That’s an unfortunate limitation, but not a deal-breaker.

This is true. The code is not presently set up to load HITEMP/HITRAN data. This is a good suggestion and we would like to extend the model to allow LBL simulations in the future.

- 6. It looks like the software hard-codes Plank mean absorption coefficients as (fourth-order?) polynomial fits? Same for WSGG. There is no discussion of why this was chosen, how the coefficients were determined, or a characterization of its accuracy. Do you observe any problematic behavior with the fourth order polynomials over temperatures from 300 to 3000 K? Are these interpolants or regression of data?

The software implements the PM and WSGG models as provided in the cited references, which use polynomial fits. Model details are noted in the cited references. Use of a polynomial fit is standard for WSGG models, and a polynomial fit to temperature for species for the PM model seems reasonable. The PM model used is the one advocated by the TNF workshop cited in the paper, which is standard in the combustion community. The temperature range is 300-2500 K for the PM model. This is now stated in the paper, and the code enforces this range. The range for the WSGG model is 300-2400 K. The range for the RCSLW model is 300-3000 K. Temperatures outside these bounds use these bounds for the WSGG and RCSLW models. This is now stated in the paper.

- 7. The two previous items combined lead to a significant limitation of RadLib: one cannot easily add new species or modify existing data that is used to generate the absorption coefficients since there has been internal preprocessing of (an unknown subset of) the HITEMP/HITRAN databases to produce curve fits. And given that this process is opaque, it isn’t reproducible.

The implemented models described in the cited literature. The novelty of the presented library is not the development of the absorption models, but the implementation, validation, and documentation of a library for application. Using RadLib to generate correlations for species using HITEMP/HITRAN is outside of the intended scope. We acknowledge that extra work is required to extend the models to additional species, or to extend the temperature ranges. However, the species included cover a very wide range of practical applications, certainly in the combustion field, and models for other species (e.g., for WSGG, and RCSLW) are often not available. Indeed, Bordbar’s WSGG model is one of the few (if not the only one) that is provided for arbitrary compositions of CO₂

and H₂O, with others assuming fixed or limited ratios. That said, radlib is documented and while the code does not provide extension machinery directly, extending to additional species by following the existing code structure would not be difficult if such data/correlations were available.

8. RadLib only considers CO₂ and H₂O for WSGG, which is fairly limited. Even for Plank mean, only H₂O, CO₂, CO and CH₄ are considered. NO and OH would be nice to include. It appears that including additional species would be a non-trivial undertaking.

See our reply to the previous comment regarding additional species. For heat transfer applications in combustion, NO and OH are minor species and will not contribute in any meaningful way. CO₂, H₂O, and soot will drive the radiation in practical configurations. It would be nice to include CO in the WSGG model, and we would like to do this following Bordbar's approach. But even CO is a minor species compared to CO₂, both in terms of its concentration in practical systems (usually), and its absorption coefficient on an equal concentration basis. CH₄ was only included in the PM model from the TNF workshop since many of the fuels used there include CH₄. But CH₄ will largely be present at relatively low temperatures, not the highly radiating flame zones.

We note that major revisions have been made to the paper text to clarify points like these for readers who do not specialize in, e.g., combustion processes or modeling and provide additional information about why these models were chosen for implementation in RadLib.

9. Documentation:
 - i. Generating the documentation directly via doxygen v. 1.8.20 gives numerous warnings about obsolete tags and then fails because it is looking for a relative path ".../.../docs/doxygen" that doesn't exist. This should be fixed.
 - ii. Generating the doxygen documentation via the build system target ("make doxygen") does seem to work fine.

The build process has been updated so that documentation can be generated by the system target or manually by running the command `doxygen` in the docs folder. We have removed obsolete flags, and also updated the source documentation to be consistent with function signatures.

Reviewer 2

A radiation spectral property library is developed in this study using C++. A Planck-mean absorption coefficient-based model, a WSGG model and a RCSLM model from the literature are implemented in the package. Verification results are presented and discussed. Although the authors addressed an important issue of radiation modeling, the contribution is marginal in the reviewer's opinion. No new models/methods are proposed in this study. As the authors correctly pointed out, the solver for radiative transfer equation is equally important, which is not included

in the library. The coupling of a spectral model to a specific RTE solver, can have different levels of challenges depending on the combination. The design of the library doesn't really address any of the RTE-related issues, nor provide any new solutions to simplify the coupling of spectral models to RTEs. If the authors can demonstrate that the library can be coupled with popular CFD software and popular RTE solvers such as P1 and DOM, it would further improve the quality of this study. A few other concerns are listed as follows:

The authors thank the reviewer for their comments and suggestions. It is true that the library presented does not develop new models, but implementing and providing those models is nontrivial, and we believe makes a good contribution that is within the scope of this software journal.

We agree the reviewer's suggestion regarding coupling with popular CFD software and a corresponding RTE solver, and that this would strengthen the paper. We have made significant revisions to the paper and code in this regard. First, we developed a new Fortran interface that calls the underlying C++ code. Second, we coupled the code to the Fire Dynamics Simulator (FDS) code, developed by NIST. FDS is a Fortran code that required the new Fortran interface. FDS has several radiation models available including WSGG DOM implementation. Radlib was interfaced to this model by using the radlib Fortran module in the source code. We then replaced the existing calls to the functions for computing the absorption coefficient and weighting factors with calls to radlib. FDS does an outer loop over the gases/bands with inner loops over the local grid points. RadLib was not set up for this, since it computes the properties for all bands at a given point. Hence, direct use of RadLib would have been inefficient. As a result, we implemented a new interface function `get_k_a_oneband`, that facilitated implementation in FDS. In the revised paper, we present results comparing the WSGG and RCSLW models in FDS applied to the FM Burner validation case that is detailed in the FDS Validation Documentation. We hope that this effort will satisfy the reviewer and improve the paper as desired.

1. RADCAL data were used to generate the Planck Mean absorption coefficient. Based on the reviewer's experience, the Planck Mean absorption coefficients can have meaningful differences when generated from RADCAL and from HiTEMP2010. It is recommended a more recent database being used for the library.

The Planck Mean absorption coefficients model as implemented in RadLib refers specifically to the radiation model in [3,5], which is one of the most common radiation models applied to turbulent combustion simulations. It was chosen because of its popularity and widespread use in combustion simulations (despite its limitations). That said, there is value in using the most current tools. Given the reviewer's experience with variance in the Planck Mean absorption coefficients as calculated from various databases, the authors will strongly consider adding an updated Planck Mean model using the most current available data. That said, the PM model implemented is generally applied in optically-thin situations with relatively low radiative transfer in which case the errors are not as significant.

2. Page 10, missing information on [INSERT RUNTIME COMPARISONS HERE].

This has been corrected.

3. In Figure 2, the RCSLM seems to consistently perform better than WSGG, except in Example S2. In addition, WSGG performs worse than PM in Example S4. Can the authors explain why?

The WSGG model coefficients presented were those from Bordbar's paper. These have since been updated and included in the revised code and we have likewise updated the simulation results in the paper. For the new results, the WSGG model no longer performs poorly for example S4. For example S2, the RCSLW model converges to the LBL results as the number of gases increases, but the WSGG model outperforms the RCSLW model for 4 gray gases. It is not clear why the WSGG model outperforms the RCSLW model for this case. We have added some relevant discussion to the paper. We do note that the WSGG model is empirical and contains very little representative physics, which makes it difficult to explain the behavior of the model.

4. Equation 6 appears to be inaccurate.

Upon review, the authors find no errors in Equation 6, which matches Equation 20.65 of the general band model (of which the WSGG model is a subset) derivation in [2] and appears to be consistent with the other equations in the paper. The equation is $q = \int I(s) \sin \theta ds$. The notation here is that I is a function of direction, but is itself a scalar so that $I(s)$ is equivalent to a vector I in direction s .

5. Typo: page 4, line 49, "relating them" -> "relating the"

This phrase has been removed in the revision.

Reviewer 3

The Computer Program in Physics (CPiP) submission titled, "RadLib: a radiative heat transfer model library for CFD" by Stephens et al reports a library of radiative property models that is modular and can be interfaced with other radiation model components. The manuscript is well-written and well-organized. The works cited are reasonable. RadLib contains a gray Planck-mean (PM) model, a weighted sum of gray gases (WSGG) model, and the rank-correlated SLW model. This is definitely an important tool for the combustion community, where the radiation modeling is often oversimplified mainly because of the lack of readily available appropriate radiation model libraries! I am not aware of any open-source libraries that combine the three property models presented here. The models included in RadLib are already well-established and well-validated in the radiation community. The library is written in C++ and is supposed to work in any Linux-like system. The code is well commented, although the associated documentation needs to be improved (see major comment #1).

Considering all these, I think that this CPiP submission is definitely within the scope of the journal and of interest to the research community. However, the following concerns must be addressed before the submission is considered for publication.

The authors thank the reviewer for their insightful comments and suggestions. As members of the combustion community, we heartily agree that a tool like RadLib could benefit researchers and facilitate radiation model development for combustion CFD.

1. I could not successfully build and run the library in two different Linux systems. In both cases, first, the cmake returned an error: “CMake Error at c++/CMakeLists.txt:34 (install): install TARGETS given no ARCHIVE DESTINATION for static library target “radlib”.” I could resolve this by adding ARCHIVE DESTINATION line on the c++/CMakeLists.txt file. But, then on during “make” my installation was interrupted several times because of missing packages in my Python installation. I would strongly recommend that the authors modify the installation instructions to include: a) list of dependencies (i.e., what compilers, what versions of python, what python packages, etc.), b) how to install the dependencies before building RadLib, c) a way to build RadLib with either only C++ or only Python. These installation notes do not necessarily need to be in the main manuscript, but they must be in the installation README of the code package.

The CMake build system has been updated such that ARCHIVE DESTINATION is explicitly specified, so that error should no longer occur. A build option has been added that makes the Python interface installation optional. Note that the C++ source code is required to install the Python interface, so there is no option to build only the Python interface. This information has been added to the README.md files located in the root directory and the build directory along with a more thorough list of software dependencies and required Python packages. We have also done full tests of the installation and running of all examples on a desktop Mac, and a desktop Linux machine. The c++ and Fortran interfaces were tested on a large Linux-based cluster. The code also runs on a code ocean module.

2. The library includes PM, one WSGG, and a RCSLW model. The authors should clearly indicate the rationale of these choices (i.e., why only these three models) early on in the manuscript. Since this is intended for researchers who, presumably, are not experts in radiation modeling, the inclusion of these three models can be misleading. One might think that these three models are the most appropriate models for their simulations.

Section 2 has been heavily revised and should better address this issue. In particular, at the beginning of section 2, we note the why these models were chosen. The three models are practically relevant, and represent a variation in complexity and accuracy.

3. RadLib is intended to be expanded with new radiation property models in the future and interfaced with a user-supplied solver module for RTE. A brief discussion on how to interface such an external solver module should be presented. There is an example of a ray-tracing solver presented in the code files, but a brief discussion on coupling one’s own RTE solver is important in the manuscript. Additionally, a brief discussion on how a user can add a new radiative property model should also be included.

Section 3 has been revised and extended to include more information on how users can include and interact with the RadLib library, including explicit references to example files that further illustrate usage of the library. We have also added a subsection to the

results discussion that describes the integration of RadLib with the Fire Dynamics Simulator (FDS) code developed by NIST. Adding additional radiative property models to RadLib would require creating another class object for the implementation. The intention would be for the new model to inherit from the `rad` base class and reimplement the interface functions `get_k_a` and `get_k_a_oneband`. Depending on the model, additional functionality may be needed. The Fortran and Python interfaces are wrapper functions that can be extended by paralleling the existing code, which is much simpler than writing these interfaces from scratch. A discussion on these points has been added to Section 3.

4. The Planck-mean absorption coefficient included in RadLib is based on TNF workshop data (from 2003). According to the references cited, these are fairly old correlations. If the authors use these data, I am concerned that the model parameters and correlations used in RadLib for the PM model may be obsolete and inaccurate. Can the authors please comment on this?

We purposely chose the TNF workshop data for calculating Planck Mean absorption coefficients because it is one of the most common and widely used radiation models applied to turbulent combustion simulations, despite its age and limitations. Section 2.1 has been revised to include a more explicit discussion of why the TNF radiation model was chosen. Sections 4 and 5 include example cases and results that demonstrate the deficiencies of the PM model compared to the other models included, and some additional discussion is given there. We intend to extend the model to include updated PM data.

5. On page 13, the authors present a paragraph describing Figs. 2 and 3 - the comparison of various radiation property models present in RadLib in one-dimensional configurations. The comparisons are presented simply as observations without any explanation or reasoning why the models behave the way as seen in Fig. 2 and 3. If I understand correctly, the focus of the work presented here is not the comparison of the accuracy of the three models but to showcase the library. Therefore, these comparisons - specifically without any explanation of the results - may not be essential to the current manuscript, and can be shortened. These results do serve as descriptions of example cases provided in RadLib. On the same page, the authors say that “While omitted [should be omitted] here for brevity, the implemented WSGG and RCLW models give essentially identical results to those presented in [11, 14] such that these examples also serve as a validation of the implementation of the models.” The authors should avoid qualitative description such as “essentially identical” in the context of validation and present a quantitative (e.g., % error) metric wherever possible. A strong validation is important for programs like this.

While it is true that the purpose of this paper is to present the RadLib library rather than evaluate or compare the models, the authors believe that example case results in Figures 2 and 3 are important. These illustrated examples RadLib’s capabilities, demonstrate the example cases provided with the code, and provide comparison between the models and with reference to the LBL data. Comparison between the models has relevance to the reviewer’s second comment above regarding the rationale of the models chosen, and our

discussion of them in reply in the text. The reviewer's point about presentation without explanation is valid. The discussion in Section 5 has been expanded. Some of the results and discussion have changed since are now using updated parameters for the WSGG model that improves the agreement of that model, especially for Example S4. This makes the models for the different cases presented more consistent, lessening some of the burden for explaining anomalous behavior, though, due to the empirical nature of the PM and WSGG models, full explanations can still be difficult.

Regarding validation, we agree with the reviewer in being careful about the language used. We are fully confident in the implementations of the models presented. Model parameters were checked and rechecked, and results from RadLib were carefully compared to the corresponding model results presenting in the original papers. In the case of the RCSLW model, which is the most complex to implement, we had the benefit of discussions with the inventors of the model (noted in the Acknowledgements), and even compared our code output to their Fortran implementation. We have modified the text to avoid qualitative description.

6. The computational cost tables for two cases show a significantly different trend, which the authors attribute to "additional overhead required in performing the calculations." Can the authors be more specific as to what these overheads may be? They say that they have neglected the initialization and input/output costs. Also, do these CPU times presented include the cost for the ray-tracing solver as well? Are the grids kept the same for all configurations?

We have modified the discussion slightly by moving the note about the cost not including model initialization and input/output earlier to make it clear that that applied to all results. Two cost comparisons are given: (1) cost of the property evaluation itself; and (2) cost of the full RTE solve (including the ray-tracing solution and property evaluation). Item (2) is relevant since property evaluations are not normally done in isolation, but part of a larger RTE solve, so that differences in the property model costs have less impact on the overall cost. This is now noted in the paper. The revision should be clear about the costs including the ray-tracing solver and the constant grids. Regarding the "overhead," the revision reorders the discussion and that word is not used. This discussion parallels what is noted here, that differences in the overall costs are smaller when accounting for the cost of the RTE solver as well as the property evaluation. (Note that Section 4.1 has since been moved to Section 5. It and the appropriate figure captions have also been revised for clearer communication of the points above.)

7. In line 53 on page 4, the authors claim to have CH₄ property in RadLib but none of the validation cases contain CH₄. Furthermore, for RCSLW, the refer to Pearson's work as a source for data of CO₂, H₂O, and CO without indicating the source data for CH₄.

CH₄ is present only in the PM model sourced from the TNF Workshop [3,5], which includes CH₄ absorption coefficients because most of the TNF Workshop target flames are fueled by CH₄. In general, CH₄ is not a significant radiatively participating species in combustion systems, but it is included in the PM model for completeness with the PM source. The revision notes differences in the models and species, and default values are

included for species in the interfaces (also now noted) so that users don't have to specify mole fractions for species that are not used in a given model.

MINOR:

1. It should be clarified that RadLib only provides radiation properties for non-scattering media.

This clarification has been made explicitly in Section 2 and now appears elsewhere the text where appropriate.

2. In the program summary the description of the nature of the program is somewhat confusing from a grammatical construction standpoint. Particularly, the following part, "This presents a problem for practitioners who wish to use/implement radiation models for which [refers to what?] such models constitute [constitute?] only a small but important part of a larger simulation. Turbulent combustion simulations are one such example [example of what?]. Often, rudimentary assumptions [in what?] are made and this can negatively impact results."

The program summary has been rewritten for clarity.

3. In Page 2, "In some cases, radiative gains or losses are small compared to other energy sources or heat transfer modes and radiation can be safely neglected, ... difficult to simulate accurately without a robust radiation model." Please provide some references to support these arguments.

This statement has been removed in the revision. In the spirit of this comment, we have added references to the text where needed to bolster weakly supported arguments. In particular, due to unrelated revisions, such discussion appears in Section 2.1 since it is most closely related to the Planck Mean model and its limitations.

4. Please explain all the symbols used in equations. For example, η in Eqn. 1 (and subsequent equations), E_b in Eqn. 11.

Text surrounding equations has been revised throughout to ensure explicit definitions of variables.

5. Page, 4: "Global models are an important class of radiation property models that make use of spectrally-integrated radiation properties and are usually versions of the weighted sum of gray gases (WSGG) model [1, 2]." This sentence is somewhat misleading. It may be more appropriate to think of WSGG models as special cases of SLW or FSK models.

The authors agree that this statement could be potentially misleading. The second half of the sentence (beginning with "...and are usually...") has been removed.

6. In lines 48-52 on Page 4, the authors say that the absorption coefficients are calculated using some correlations after some curve-fitting to spectroscopic databases. I am not sure

what the authors refer to as correlations. There may be some correlations involved in WSGG models, but it is not clear to me where are correlations involved in RCSLW.

The absorption coefficients for the gray gases are developed from high resolution databased, and correlated, for each gray gas considered, as functions of the local gas state (temperature, pressure, and composition). The PM model uses a temperature correlation for each species absorption coefficient, with the overall absorption coefficient computed by combining the species. The WSGG model uses correlations developed for the gray gases based on emissivity calculations from the high-resolution databases, and treats the composition through the molar H₂O/CO₂ ratio. The RCSLW model's key tool is the absorption line blackbody distribution function (ALBDF), which is calculated using spectral data available from databases like HITEMP. See Solvjov 2017 (cited in the paper) for details. The revision includes such discussions for the individual models. The modification of the original statement is in the last paragraph in Section 2 before section 2.1.

7. On page 7, the authors mention that the PM model can be “reasonably accurate” in some cases and go on to say that, “The Planck Mean model is most appropriate under optically thin conditions with relatively low radiative transfer relative to other heat sources such as reactive heat release rates.” The Planck Mean model will be appropriate in optically thin cases as there is very little reabsorption. This may or may not be connected with “relatively low radiative transfer relative to other heat sources.” In fact, if the radiative heat transfer is much lower than chemical heat release, neglecting radiation altogether may also be reasonable! Although, the definition of “relatively low” is subjective.

This discussion is modified in the revision. The PM model is not directly connected to optical thickness of flames. In combustion applications, the PM model is often used in situations that are optically thin. In such cases, the radiative losses are relatively low (often around 5% for laboratory jet flames), and use of the PM model may provide sufficient accuracy given that errors are active at the 5% level of the overall radiative losses, so not severe. When coupled with the optically thin assumption for RTE treatment, the radiative solve is relatively simple and inexpensive.

8. On the same page, lines 40-41, “This version of the WSGG model is advantageous because it allows for arbitrary CO₂ and H₂O compositions, increasing its accuracy and flexibility.” Can the authors provide some references that show the increased accuracy and/or flexibility of this WSGG over any other WSGG?

Two references for the chosen WSGG model are given earlier in the paragraph. The second (Bordbar et al. 2020) includes detailed comparisons between this model, its immediate predecessor (see Bordbar et al. 2014), five other WSGG models, and line-by-line calculations for four different test cases that show high accuracy over a wide range of molar ratio values. We have noted this in the revision. While other models may have performed better in individual tests, this model demonstrated consistent accuracy over a range of scenarios. For clarity, this reference has been added to the end of the aforementioned sentence so as to better indicate where to obtain relevant information.

9. With respect to Eqn 8, M_r is defined as molar ratio y_{H_2O}/y_{CO_2} . Are the authors using y as a symbol for mole fraction rather than mass fraction as is conventionally done?

We have changed the notation in the paper to use x instead of y . This is consistent with our use in the code, and we also prefer x for mole fraction and y for mass fraction. Solovjov (RCSLW) and Bordbar (WSGG) both use Y for mole fraction though.

10. Please remove the words “[INSERT RUNTIME COMPARISONS HERE]” on Page 10.

Done.

11. In lines 29-30 on page 13, please insert the units for Q in -80 and 200.

Done.

12. In the Discussion (page 16), the authors write, “Neglecting radiation or using simple models like the optically thin assumption is adequate for some cases with simple geometry or limited chemical reactions...” This can be misleading. I am not sure if either the simple geometry or the limited chemical reactions give any indication of the importance of radiation. Rather, something like a radiant fraction or optical thickness may be a more appropriate metric to justify neglecting radiation or using an optically thin model.

This is a good point. The revision should correct this. We partially answer this in reply to your comment above regarding page 7 that the “PM model can be ‘reasonably accurate’”. In particular, we are revising the discussion around the PM model to note that it is not an overly accurate model in comparison to the others, but it is widely used, and its use is warranted in situations with relatively low radiative losses that may occur, for example, in cases where radiative losses are low compared to heat release rates, or where flow timescales are low compared to radiative transfer timescales.