Reply to reviewer 1 for Silicone

Many thanks for your detailed review. You raise several excellent points and we have significantly reworked some parts of the paper to address these. We present this summary in the format:

- Original comment
    - Our response
        - Text change, [structural change]

Response to overview comments

- It could be helpful for users to briefly mention the package infrastructure used by silicone and if and how it is connected to the other packages in the IAMC toolbox (e.g. data transfer using a common csv format, or by API calls to other packages)
    - We have expanded the section discussing the pipeline to mention data transfer. The section now reads:
        - The pipeline is based around the pyam package (Gidden and Huppmann, 2019), specifically its IamDataFrame class, which Silicone makes extensive use of. Pyam dataframes easily convert from and to widely-used pandas dataframes, which pyam and Silicone also use internally (McKinney, 2011). The pipeline also includes tools to harmonise (i.e., correct projection made in the past to match now-known emissions) (aneris, (Gidden et al., 2018a) before infilling and to pass the complete projections to climate simulators. The estimation of climatic impact is performed by OpenSCM,
- For me an overview over the processing steps would be great.
    - We describe all the processing steps in the Methods section. Other than getting the data into pyam form and harmonising it (if desired) there aren't any other pre-processing steps needed. We've added a description describing the interaction between silicone and the harmonisation process in the mathematical detail section. Is there something else you would like to see discussed?
        - If the results are to be harmonised, then harmonising both the infiller and target data before infilling is required for improved consistency (otherwise infilling depends on outdated data). Absolute value infilling techniques preserve harmonisation, however ratio-based approaches do not necessarily, and may need harmonisation again afterwards.
- It would be good to have more structure in explaining the different crunchers and infilliers (e.g. paragraph headings for each cruncher) The paper will also be used as a reference for silicone and then it's good to easily find information on a specific issue. The information on crunchers and infillers is somehow scattered through the document with bits in the methods section (text and tables) and some aspects in the results section.
        - [Sub-subheadings have been added to the sections detailing the crunchers]
        - [The rank correlations analysis has been added to the results section. Although not formally a result of silicone, this hopefully provides a clearer separation of model description and data.]
- The overview tables are a good idea, but they should be placed in the methods section for the print version of the paper.
    - We agree, but have no control over typesetting – tables were requested to be at the end of the document at this stage. Hopefully the editors will take note!

- It is a bit unclear how robust the results are and how much work is needed to check them before using the results. It is mentioned in some places, especially the tables, that incomplete databases can influence results. For easy use of the package it would be great if the user is warned in these cases. As one purpose of the package is to fill those gaps it would be a problem if the results are unknowingly influenced by the gaps. Maybe you could add a short section on limitations which gives an overview over such possible problems and references the tables for details. See also comments on lines 99-103 and lines 130-133.
    - The code has now been updated so that the default behaviour is to ignore inconsistent data (e.g. scenarios missing values at one time). There is also now a warning when infilling negative lead emissions with the time-dependent ratio.
        - [The text for “time-dependent ratio” and “decompose collection with time-dependent” ratio has been changed significantly to reflect this.] “This relies on all scenarios having values for all of these variables, so misses out cases which do not have one of the constituents or only reports at some of the required times, unless the override option “only_consistent_cases” is set to False.”
- The gas basket splitting functionality seems to be not fully developed. The package has the potential to replace the EQW and it’s update used by assessments like the Climate Action Tracker, but in it’s current state it can not yet do that. It would be great if using QRW or EQW with a KyotoGHG constraint was possible for all gases. Could you add information if/how this can be added e.g. as a multiple infiller or a scaling postprocessing step? This would make the package very useful for the climate policy assessment community.
    - Since writing the first draft we made another multiple infiller that does exactly this. We have now included details of it in the paper. Note that one of the gases is infilled by the conservation condition
        - [“Split collection with remainder emissions” entry added to multiple infiller table and flow chart, described the process of infilling differently for breaking up the aggregate values into their components]
- RMS closest: please consider referencing the paper "Warming assessment of the bottom-up Paris Agreement emissions pledges" (YR du Pont, M. Meinshausen. Nature Communications, 2018) where a similar method has been used
    - Good suggestion. Included as follows:
        - The alternative approach of inputting the whole pathway with the smallest mean-squared distance over all time was used in (Robiou du Pont & Meinshausen, 2018). This works well for large databases containing similar paths, but is less useful for smaller databases or for paths with an unusual behaviour over time.
- It is very useful that the authors test the correlation between the different variables. The Spearman’s rank correlation coefficient does not detect nonmonotonic relationships which can be modeled by the Quantile Rolling Window method. Have you tried other methods to detect correlations (e.g. the Hoeffding Dependence Coefficient).
    - As mentioned in the previous correspondence, we have tried only methods that would also not detect non-monotonic trends. Correlations for all variables with CH4 and CO2 were also plotted and inspected by eye for three times and no clear nonmonotonicity was seen, however.
        - We also plotted the relationships between CO_2 and all other variables (using the plotting function in the Silicone examples github) to check that there were no obvious cases of a non-monotonic relationship.
- • Did you investigate non-emissions lead variables?

    - No, although the mathematics should work identically. We now point out using economic indicators as a potentially use-case for ratio infillers, although strictly speaking it's not "infilling" if this data is used (it's just a very basic IAM).
- Have you used the method on other emissions databases (AR5DB, SSPDB, ...). Is this easily possible or does the package need a structure only present in the SR15DB?
    - As described below, we now apply Silicone techniques to infill a model/scenario from AR5. It is also being applied to AR6. Given that all these listed databases use the IAMC format (more or less), they are easily handled by pyam and hence Silicone. The only pinch point for using other databases is getting the data in the pyam structure. Many databases are in pyam or pyam-compliant structures, so following on from the point below we now show infilling a scenario from such a source.
- Finally a more political comment. Why do you feature a fossil fuel companie's scenario in a climate change paper (Shell sky scenario)? This helps their greenwashing attempts. Please provide the scientific explanation for using this scenario.
    - This was originally referenced because that scenario is claimed to be Paris-compliant and was also one the industry scenarios included in the IPCC SR1.5 pathway assessment. However, it is our contention that this depends on the way that the scenario is infilled. However since to demonstrate this point would require careful data handling, harmonisation and running of a climate simulator to show convincingly we will leave that discussion for elsewhere.
        - [The Sky model infilling has been replaced by POEM scenario B infilling from AR5. Text and images have been updated to show this.]

Detailed comments – note that line counts are drastically different now

- Abstract: I think it would be good to mention some of the most important use cases and infilling options in the abstract.
    - Good idea, we have added a sentence
        - We demonstrate the package's utility with three examples: infilling all required gases for a pathway with data for only one emission species, splitting up a Kyoto emissions total into separate gases and complementing a set of idealised emissions curves to provide a complete, consistent emissions portfolio.
- Line 70: It would be helpful for the reader to have short description of the structure of the paper at the end of the introduction.
    - Good idea
        - This paper is structured as follows: the *Methods* section presents an overview of the different infiller methods, then goes through the infiller techniques in precise and mathematical detail. In *Results*, we present our analysis of emissions projections in the SR1.5 database. This includes correlation statistics of the database, and how well Silicone reproduces aspects of it from the rest. We use this to draw conclusions on the implications for using Silicone on unknown data. In *Use Cases*, we present three examples of using Silicone for infilling a pathway with limited information, splitting up an aggregate basket of emissions and infilling stylised emissions trajectories. We end with a summary of our paper.
- • Line 83: Please give an overview over the infilling process at the beginning of the methods overview section or by expanding the introduction of the methods section (i.e. what are the steps in the process).
    - This is now included, if you mean this in a philosophical sense. We also now discuss elsewhere the protocol for harmonisation. If you mean in terms of which commands

to use in what order, that seems best learnt from code examples, found in the notebooks.
        - Silicone offers a range of tools that apply methods for doing this infilling which are appropriate in different circumstances, depending on the amount of complete data and how much we know about the narrative behind our emissions. These tools are referred to as 'crunchers'. Each of these crunchers takes a 'lead variable', found in both the infiller and target databases, and uses it to infer the value of a 'follower variable', found only in the infiller database (hence missing in the target database). There are also several tools for easily infilling multiple variables, called 'multiple infillers'. These may have multiple follower or lead variables.
        - If the results are to be harmonised, then harmonising both the infiller and target data before infilling is required for improved consistency (otherwise infilling depends on outdated data). Absolute value infilling techniques preserve harmonisation, however ratio-based approaches do not necessarily, and may need harmonisation again afterwards.
- Lines 99-103: Does the code have any flags to control behavior in these cases? Does it warn if this occurs or is it up to the user to control the results?
    - A warning is now reported for ratio methods with negative target leads, although it is up to the user to determine if the use is acceptable or not.
- Line 119: Where do the estimates averaged over come from? Different historical estimates for the same year, or different years as well? Is it possible to use trends of the last years with data or averages over the last years? If only single last historical data points are used (which is what I understand from table 1) the method is very sensitive to annual fluctuations in emissions data. Thus trending or averaging is needed for robust results that do not drastically change by using one additional year with data that has unusual emissions due to e.g. extreme weather or crises (financial, COVID-19, local crises, ...).
    - The estimates are not averaged over any period. The vast majority (in our case all) of projected emissions trajectories data reported by integrated assessment models is available at 5 or 10 year intervals. Such data points therefore aim to represent systemic changes rather than (sub-)annual effects. Still, in the case of emissions that fluctuate strongly over time, this infiller should be regarded as less reliable. In the manuscript, we describe the potential use case of this infiller, and we now have added this pitfall in Table 1.
- Line 123: I assume "cases" refers to scenarios and scenarios where the sign for the lead variable does not coincide with the infillee are taken out of the average for both lead and follower variable? Maybe revise language to clarify that. A more general point: it's a nice way to avoid numerical problems at transitions to negative emissions. However, when working on small scenario databases and / or with high temporal resolutions you might run into problems with availability of scenarios that use the same time period for the transition to negative emissions as the infillee scenario. What do you do in those cases? Does the algorithm allow for a fallback option or does it just fail?
    - You assume correctly (rephrased for clarity now). In this case, the infiller hard fails. The user should decide explicitly whether to use the sign-independent version or another infiller.
        - It will produce an error if there is no data with the required sign.
- Lines 130-133: As a lot of scenario databases are not fully consistent regarding data completeness (and sometimes sector definitions) it is dangerous to not automatically check for completeness or at least have an optional filter that removes incomplete scenarios.

    - This is indeed somewhat dangerous. We have now changed how this works so that by default, data that lacks either some of the constituent variables or only reports at some of the required times is removed before infilling.
        - This cruncher is the foundation for the 'decompose collection with time-dependent ratio' multiple infiller. It relies on all scenarios having values for all of these variables, so misses out cases which do not have one of the constituents or only reports at some of the required times, unless the override option "only_consistent_cases" is set to False. It always constructs a new, consistent version of the aggregate variable in case different modellers used different conversion factors in the infiller database.
- Lines 138: Does "with all the same times" mean data availability for the same points in time? The use of "times" seems unusual and is not intuitive to understand for me.
    - It does, rephrased
        - The 'RMS closest' cruncher filters the infiller database for models with data at all the times found in the infillee data.
- Line 142: Line 140 states that the average is taken over all points in time, yet line 142 states that "the value of the lead at one time impacts the whole timeline". That sounds contradictory to me. Can you explain? Also, adding a formula would help.
    - Rephrased, formula added
        - The 'RMS closest' cruncher filters the infiller database for models with data at all the times found in the infillee data. It then ranks models and scenarios by the root mean squared (RMS) difference between the lead data in the infiller and infillee database, with the average being taken over all timeslices. It returns the follower data from the scenario/model combination with the smallest RMS difference: the formula is $E_f(t) = e_{f,i}(t)$, where the subscript *i* refers to the model/scenario case that minimises $\sum_t \left(E_l(t) - e_{l,i}(t)\right)^2$. In the case of a draw, the value that occurs earlier in the infiller database will be used. This is the only cruncher that is not time-independent, i.e. changing the value of the lead at one time may result in different outputs at other times.

- [Many comments about QRW] If I understand correctly, the windows are not windows with clear boundaries but created by the weighting functions. I think the phrasing "Five windows are drawn" is a bit misleading because to me it suggests that the data-points are binned. However, my understanding is that you actually create a smooth distribution from each discrete data point such that at every lead emissions level chosen to calculate a quantile you have all data follower emissions
data-points available, just weighted by the weighting function.
    - Mostly correct, one other possible misconception is that we default to 10 windows, 5 was just chosen for illustrative reasons.
        - A number of rolling windows centers (here 5, by default 10) are drawn and a weighting function constructed for each window. It has a continuous distribution, rather than a discrete cutoff, hence the name.
- Please introduce an index for the data points used for the analysis.
        - [We have significantly changed the notation for the formulae to do this.]
- why do you use the approach of in-terpolating between only 5 points. When calculating the quantiles each time you use the method, you could just calculate them for the exact value of the lead emissions. If you pre-calculate the quantiles and use a lookup table to fill emissions (as done in the old EQW), you could use a higher resolution than just 5 points. I get the

impression that you kept some aspects of the Generalized Quantile Walk that are no longer necessary with your improved method, because you don't have the binning.
    - The five points we have used are mainly for plotting clarity. It is up to the user how many points they wish to use and they could use arbitrarily high numbers of points (we default to 10 because, in practice, there is little variation at higher resolution than this). Another reason for using fewer pointsis computational simplicity – calculating a quantile at each point is O(N) in the infiller database, so doing so for each point in the infillee database can get time-consuming for larger infillee databases.
- Line 149: Maybe add "for each time step" to clarify that the operation is carried out at each time step individually. This information could also be added in the description of the process in the following lines.
        - We have added "for each time."
- Line 149: If I understand correctly you don't interpolate between quantiles, but between follower emissions associated to a given quantile at given lead emission values. If that's correct, please rephrase the sentence.
    - That was indeed not the interpolation we were trying to talk about here
        - infills the values based on interpolating between the quantiles of the follower variable.
- Line 155: There is an additional full stop.
        - [At some points, the whole text of the caption was used rather than just a reference. This has been deleted]
- Line 162: The formula uses ri as sum index, but neither ri nor rp are introduced in the text. (I assume ri are the lead emission data points in the scenario database ordered by the associated follower value and rp is the highest one to be taken into account at quantile q (so should it not be rq instead of rp?))
        - [Significantly rewritten in accordance with the above]
        - $q(e_l(j)) = \sum_{e_f(i)<e_f(j)} w_p(e_l(i)) + \frac{w_p(e_l(j))}{2}$.
- Line 181: Why did you choose 0.03?
    - To select only extreme cases.
        - We also calculate the variation of this value with time, and in cases where this exceeds 0.03 (chosen to highlight only extreme cases),
- Line 215: Did you analyze if the RMS closest cruncher chooses scenarios coming from the same model or study such that chosen pathways are very similar?
    - We didn't in detail, however they typically this won't happen as the same model will have a large spread in well-modelled variables between the scenarios by construction (there's little point in repeating yourself). By definition, the RMS closest cruncher will choose similar pathways in terms of trend.
- Line 252: The sentence starting with "that" is incomplete.
    - Again, a figure-caption error
        - [deleted]
- Line 256: Does the EQW use interpolation between quantiles and smoothing before calculation of the quantiles?
    - We have added have a section explaining this properly, as follows
        - The equal quantile walk calculates the quantile of the lead value at each time. This 0 for values below the database minimum, 1 for those above the database maximum and the fraction of infiller data smaller or equal to this

value otherwise. We interpolate between neighbouring values in the infiller data to avoid rounding errors.

- Lines 259-262: Did you test the influence of the weighting function on the conservativeness of the QRW cruncher. I assume that in the extreme case of constant weighting functions this would always return the median of the scenario database. So the smoothing through the weighting function has to be used carefully.
    - We did, you are correct.
        - Increasing the decay length will reduce the weight difference between points, so the rolling window becomes wider and more even, with the limit case of calculating quantile q of all data for large d_l.
- Line 275: See also comment on line 123. How many scenarios are left when restricting to scenarios with same sign. This will especially be problematic for extreme stylized scenarios with early negative emissions.
    - Indeed, see comments above.
- Line 279: a similar approach has been taken in "Warming assessment of the bottom-up Paris Agreement emissions pledges" (YR du Pont, M. Meinshausen. Nature Communications, 2018)
        - [Reference added]

Comments on testing the examples repository

- There are problems with directory names. The input data directory in the repository is called "input", while in the code is is referenced as "Input" with upper case "I". This leads to the code failing on unix systems.
        - [Fixed]
- Splitting up basket: The notebook fails in cell 7,line 8 with: "TypeError: convert_unit() missing 1 required positional argument: 'to'""
    - This problem should not happen with the latest version of silicone, do you still find it?
- Stylized Path: the legend of the plot resulting from cell 9 shows 4 cases of MESSAGE scenarios, yet in the plot only two can be distinguished. Is that correct?
    - Correct, E_∞ does not appear in this formula so two of these lines are identical.
- Is it possible to store calculated quantiles for the QRW and EQW method to increase calculation speed by using a lookup table?
    - The quantiles for QRW are calculated at the point of generating the infiller, so several different target databases can be infilled efficiently in this way, but it doesn't speed up our use case here. More lookup tables are possible in principal but this would not demonstrate the use of this toolkit in practice though. We could add such functionality in future but we don't believe it is necessary to illustrate the tool in this paper (pull requests and issues at the github repository are most welcome). We have reduced the computational load of the new POEM infilling notebook relative to the Shell notebook by removing most of the variables, which were never plotted or explored anyway.

Comments on tables and figures

- Table1: latest time ratio: see comment on line 119. Please explain what re-harmonizing means in this context.
    - [Comment has been removed here]
- Table1: Time dependent ratio: when using the optional filtering for same sign, how do you ensure that there is a sufficient number of scenarios available?
    - We raise an error if there are none of the correct sign, It's not clear that there's a "sufficient number" required above 1. There will now be a warning thrown if the target data is negative, irrespective of the number of negative infiller scenarios.
- Table 1: Linear interpolation: Is it possible to use scenario binning here to not select two very similar pathways (e.g. same model and storyline with small parameter variation)? I think this could be made more stable by using more than two scenarios, but the problem will not vanish. Can you give an example, where using this cruncher makes sense?
    - This cruncher is only intended for use on a very limited number of scenarios. We tend to want cases where the scenarios have similar model and storylines. This then ensures that the infilled result is compatible with those storylines. E.g. my model assumes a SSP5 world and is comparable to these groups of scenarios with high biomass burning – I want to infill the amount of BC corresponding to this much CO2. We also now mention that this is very similar to a previously developed interpolator that used cubic spline interpolation, which is fairly similar in the limit of a large database. This specific cruncher in Silicone was recently used in the paper *Current and future global climate impacts resulting from COVID-19* https://www.nature.com/articles/s41558-020-0883-0
        - A tool for infilling was provided with (Rogelj et al., 2014) using a cubic spline between specific points in a small database, however this type of infiller behaves chaotically when applied to large databases incorporating many different models. It was also coded in Excel, limiting the ease of open-source development.
- Table1: EQW: Does the EQW cruncher use smoothing through a weighting function when calculating the quantiles?
    - EQW is smooth but does not have a weighting function
        - [Section on EQW added elsewhere]
- Tables 1 and 2: please repeat the table header after a page break
    - Good suggestion. This would be a matter for typesetters. Tables do not behave properly with track changes so we will not attempt to do this by hand here
- Fig 2b: there are some artifacts around the subfigures (partly visible plot boundaries?)
        - [hopefully fixed now]
- Fig 3: The spike in the linear interpolation pathway shows that this cruncher is very problematic.
    - Correct and intentional. This is partly alleviated in the current version, but we do not recommend using linear interpolation without filtering the data down carefully, as is described in the text.
        - We see from **Error! Reference source not found.** that the linear interpolation model (without filtering the database) provides a chaotic pathway, due to its value being determined only by the two points either side of it in the database, which changes semi-randomly with time and should not be used here.
- Figs 4-6: please add a legend
        - Added legends to the figures.