CEMAC priorities within A-CURE

Overview:

Within A-CURE Leighton Regayre (LR) will be creating a perturbed parameter ensemble (PPE) of several hundred UKESM (release version) simulations. The UKESM needs to be developed to include code changes identified as priorities in recent analyses, as well as code changes already implemented in our existing PPEs (in branches of vn8.4 of UKCA). A rose suite will need to be created to allow for parameter perturbations to take effect through Rose-Cylc (previously implemented using UMUI hand edits).

The effects of structural model developments (code changes) on model diagnostics need to be tested and compared to earlier versions. Additionally, the effect of each individual parameter perturbation needs to be tested in isolation prior to creating the PPE. Existing code for summarising and comparing output between simulations needs to be extended for use in our analysis of structural developments and individual parameter perturbation tests. Finally, the PPE output will need to be converted into a format ready for analysis, as with previous PPEs.

Four months of CEMAC time has been allocated to the A-CURE project, of which around 3.4 months FTE remain. We think the following tasks (in priority order) will best utilise the combined CEMAC skill-set. The combined upper estimates of tasks A-D will require around 4.6 months FTE so cannot all be completed if upper estimates are correct. The combined lower estimates of tasks A-D will require around 3.2 months FTE. Tasks E and F can therefore only be implemented if tasks A-D require much less time than anticipated.

<u>Task A:</u> **Create a Rose suite for automating the implementation of up to 50 parameter perturbations**

<u>Details:</u> Creating an individual A-CURE PPE member requires a Rose-Cyle suite through which up to 50 parameter values can be set (within multiple config files) simultaneously. This process will need to be repeated with different parameter value combinations around 200 times, once for each PPE member. In this task CEMAC will create a Rose suite for creating individual ensemble members. Ideally, CEMAC will also develop the capability for LR to automate the creation of several hundred such suites.

In creating our GASSP and UKCA PPEs we used scripts that can modify multiple hand edits and assign values to parameters from an experimental design table. In some cases the parameters were not able to be perturbed through hand edits until they were defined in the model with minor code changes. The extent to which parameter perturbations can be directly applied through Rose-Cylc is unknown.

We previously perturbed the magnitude of anthropogenic emissions by scaling the values read in from an emission inventory. In the A-CURE PPE we will perturb emissions regionally. The AQUM team may have done something similar, so we may be able to utilise their Rose stem test suites. Otherwise, regional perturbations will require code changes to alter the handling of emission inventories. Note: One alternative solution may be to have distinct emission inventory files for each parameter perturbation and then alter the pointers to emission inventory files alongside other perturbations within the Rose suite.

Outcome: PPE members (combinations of around 50 parameter perturbations) can be created by LR using Rose suites. Ideally, this process will be automated for the creation of around 200 PPE members.

Time allocation: *Around 20-25 days*. The lower estimate is based on most parameters being implemented in a manner similar to our existing UKCA PPEs, where differences in existing trunk and branch codes will inform this work. The upper estimate is based on the possibility that our experimental design includes parameters that require new code changes.

Latest completion to fit with A-CURE milestones: Feb and Apr 2019.We will be testing the effects of one-at-a-time (OAT) parameter perturbations experiments (where individual parameters are perturbed to extreme values) during the 1st quarter of 2019. The OAT simulations need to be created using the Rose suite because this will confirm the correct implementation of the parameter perturbations in the suite. We already have the necessary information for some parameter perturbations and OAT tests. The Feb completion time represents the implementation of known parameter perturbations in the suite. However, the details of other parameters will not become clear until an elicitation exercise (scheduled for the first quarter of 2019) is completed. The PPE will be created in the 2nd quarter of 2019, at which point all parameter perturbation options will need to available in the suite, hence an additional April completion time on this task.

Task B: **Implement UKESM code changes to address known structural deficiencies**

Details: Several model development priorities have been identified in our A-CURE analyses that need to be addressed if we are to maximise the usefulness of the A-CURE PPE. A summary of proposed changes can be found here: <https://www.see.leeds.ac.uk/redmine/public/projects/a-cure_ppe/wiki/Structural_Changes_(proposed)>. These structural deficiencies are not bugs in the model, rather they are ways in which the model poorly represents physical processes leading to biases when compared to observations. In several cases, code has already been implemented in GLOMAP or UKCA and can be provided. These code changes will require the creation of model branches from the trunk reference source code for the release version of UKESM. For example, the pH of cloud droplets is currently fixed globally and our analysis shows a need for it to vary regionally. The simplest code changes, with first order benefits, would be to set cloud drop pH according to a land/sea mask. More sophisticated code changes for cloud drop pH may be limited by code dependencies in modules being run in parallel. CEMAC could investigate the possibility of implementing more sophisticated structural changes. NOTE: Branches created in this task may clash with branches created in Task A. We are currently investigating which model structural errors have already have been addressed by the UKESM development team.

Outcome: The A-CURE PPE will be cutting-edge and hence will serve as an invaluable research tool during the next model development cycle. Creating the PPE without addressing known model structural deficiencies would mean that the PPE would lose analysis value as each of these deficiencies are addressed (by us or others). *Note: The code changes are to be implemented in our private PPE-specific branch only, so the rigorous UM systems team quality assurance process can be avoided at this stage.*

Time allocation: *Around 50-70 days.* The suggested time allocation is highly uncertain. Because the UKESM has been in development throughout 2018, and because this development has been informed by our ongoing analyses, it is as yet unclear what structural errors have already been addressed in the soon-to-be-released version of the UKESM. Strategic discussions around the ways to address remaining structural errors are ongoing and involve experts at other institutions. Additionally, there is some flexibility in the degree of sophistication with which structural deficiencies are addressed, which will only be decided once preliminary investigations into code structures have been undertaken as part of this task. These considerations mean we are unclear how much time will be required to implement the necessary structural changes to the model. The lower estimate assumes that several important code changes (identified using HadGEM-UKCA vn8.4) have already been implemented in the release version of UKESM, and assumes that many of the remaining code changes closely resemble those already implemented in GLOMAP and/or UKCA.

<u>Latest completion to fit with A-CURE milestones:</u> Feb and Apr 2019. All code changes need to be implemented before the PPE is created in the 2nd quarter of 2019. Ideally code changes would all be included in the model version used for OAT experiments which will be conducted in the 1st quarter of 2019. However, there is flexibility in the number of code changes to be implemented in this task, so the model version available in February (with some code changes) can be used for OAT experiments. Other code changes implemented by April 2019 can be used in the actual PPE (with implications for timings of task A).

<u>Task C:</u> **Create/adapt code for post-processing PPE diagnostics**

<u>Details:</u> Individual diagnostics from our PPEs need to be compiled into netcdf files with an 'ensemble member' dimension so that we can streamline and automate aspects of our statistical analyses. Diagnostics with multiple temporal averages need distinct netcdf files. James O'Neil worked alongside Masaru Yoshioka to create python code (for use on Jasmin) for compiling two key PPE diagnostics (Aerosol optical depth and cloud drop number concentration) at 3h, daily and monthly mean frequencies. <https://github.com/cemac/A-CURE/blob/master/Python/pp2nc_AOD550_CDN.py> Ideally, this code will be adapted for the A-CURE PPE (if required because of changes to diagnostic output format) and extended to handle many more diagnostics with different dimensions. The code is designed for use with '.pp' files. However, the output for the A-CURE PPE may already be in netcdf format, depending on the methods used to create it. Note: this task could be combined with task D, where ensembles of diagnostics will already be read for plotting purposes.

<u>Outcome:</u> LR will make extensive use of the code to process key diagnostics for further analysis.

<u>Time allocation:</u> *Around 10-20 days*. The lower estimate here is based on the assumption that the existing version of the code does not need to be significantly adapted for output from the A-CURE PPE. Also, it assumes many of the key diagnostics have the same dimensions (which will allow for greater generalisation) and that this task can be merged with task B. However, a preliminary review of the code suggests there are aspects (such as file naming conventions) that were not designed for general use.

<u>Latest completion to fit with A-CURE milestones:</u> Mar or Jun 2019. The PPE will be created during the 2nd quarter of 2019 and will take some months to create. However, the post-processing code resulting from this task could be tested on model output from the OAT tests to be completed in the 1st quarter of 2019. Analysis of the PPE will depend heavily on this code.

<u>Task D:</u> **Expand functionality of existing python plotting code**

<u>Details:</u> The aerosol team at Leeds have python code (largely created by Jesus Vergara) for routine plotting of output from UKCA simulations. We also have distinct code for quantifying aerosol budgets from model simulations. Currently the plotting code produces maps of monthly mean output for individual model simulations. There are several developments that are essential for this code to be used optimally in A-CURE: 1/ include maps of differences and ratios between output from two versions of the model (before and after implementing code changes and/or altering parameter values) as subplots on each output page, 2/ accept multiple model identifiers so that the code will run across an ensemble of simulations, rather than a single simulation, 3/ include plots of zonal means of key model diagnostics, 4/ include a table of aerosol burden summary statistics, adaptation so the code runs on LOTUS and 5/ include an option to select temporal resolution for output (3h, daily, monthly). In addition to these priority code developments, the code would be of wider use within ICAS if it were designed so that all figures and burdens could be produced for a comparison between models (such as GLOMAP and UKESM), rather than for a comparison between versions of the same model (as will be the case in A-CURE using the UKESM).

<u>Outcome:</u> LR can quickly produce analysis plots and budgets during the testing/development phase of A-CURE. Using these figures the A-CURE team will be able to identify unexpected outcomes from implemented code changes and/or from individual parameter perturbations in a timely manner, allowing for adjustments to be made before the A-CURE PPE is created. More widely, model developers will benefit from a routine comparison plot code.

<u>Time allocation:</u> *Around 15-25 days*. Around 2 days will be needed to develop an understanding of the functionality of the existing plotting and aerosol budget calculation codes. The lower estimate is based on only implementing the core requirements (numbered 1-5) and assumes that there is substantial overlap in the implementation of different aspects of these requirements. The upper estimate is based on the assumption that implementing each of the core requirements causes unique coding challenges, such as complex manipulation of pandas data frames within Python. Adapting the code so that output from different models can be compared will be possible if implementing the core code changes is towards the lower end of the estimated time allocation.

<u>Latest completion to fit with A-CURE milestones:</u> Mar or Jul 2019. Ideally, this plotting code will be used to analyse the effect of code changes (that address structural deficiencies) and OAT experiments, all of which will inform the PPE design. Since the PPE will be created in the 2nd quarter of 2019, this task will be of greatest benefit if it is completed by March 2019. However, the most important use this code will be in analysing the PPE itself, hence a July completion date (after post-processing) will still benefit to A-CURE.

<u>Task E:</u> **Develop a model/measurement comparison suite**

<u>Details:</u> ICAS researchers currently lack tools for consistently validating model performance by comparing model output with measurements. Model performance validation is especially important within A-CURE because of the number of model development tests and one-at-a-time parameter perturbation tests that need to be assessed. Ideally, this task will produce a script for plotting maps of key diagnostics (to be decided, but likely including a subset of GASSP measurements, droplet number concentrations, top-of-atmosphere fluxes, precipitation rates and satellite measurements averaged over stratocumulus zones) overlaid with measurements (from available datasets). Additionally, some form of summary information (overlaid on the map, or included in the title) that

indicates when a model diagnostic is outside of accepted measurement tolerances (to be provided) is required. This task complements task D and could be included as an option within that code.

Outcome: The A-CURE team will be able to examine model performance across a large number of simulations with far greater efficiency. We will identify parameter values and combinations that are inconsistent with measurements at an early stage, allowing us to improve our experimental design and make better use of our HPC allocation. The suite will be available for use within ICAS, so all UKESM model users. In particular, less experienced model users will benefit from the tolerance information provided.

Time allocation: *Around 10-15 days*. The lower estimate is based on all measurement data and tolerances being provided in the format required for plotting. Note: this task requires the A-CURE team to provide a list of key diagnostics for comparison.

Latest completion to fit with A-CURE milestones: Mar or Jul 2019. The greatest benefit from this task will be in Mar-Apr. The OAT experiments will be created in the first quarter of 2019 and this task will provide insight into observational plausibility of model output at extreme parameter values. The task will also provide benefits for analysis of the PPE itself. The PPE will be created in the 2nd quarter of 2019 and will immediately be processed and analysed. **NOTE: The overall time allocation makes completion of this task unlikely within A-CURE.**

Task F: **Adapt existing Met Office 'stochastic restart' code for failed simulations**

Details: Around 1/8 (12.5%) of our UKCA PPE ensemble members failed because we used 'minimal nudging' which caused model variables exceed internal thresholds and crash the model. Met Office colleagues significantly expanded the size of their most recent free-running PPE by restarting failed jobs subject to stochastic perturbations of key meteorological fields. It is our understanding that this method is currently set up to run on individual jobs on Monsoon. In this task CEMAC would produce an equivalent code for use on ARCHER that acts on multiple jobs (possibly with different start times) simultaneously.

Outcome: More efficient use of the A-CURE HPC allocation. Number of useful PPE members increased by around 10% leading to more robust statistical analyses and conclusions.

Time allocation: Open-ended. *At least 10 days*. We have not had access to the stochastic perturbation code, so have not checked its quality and flexibility, and cannot critically assess the expected time allocation needed to complete this task.

Latest completion to fit with A-CURE milestones: Apr 2019. The PPE will be created during the 2nd quarter of 2019 so for this task to be useful it must be completed by April. **NOTE: The overall time allocation makes completion of this task unlikely within A-CURE.**

TASKS NOT INCLUDED BUT IMPORTANT FOR THE A-CURE TEAM TO CONSIDER:

Parameter perturbations. Originally planned to implement parameter perturbations using Steven Pickering's pearl script. However, Steve's scripts applied to UMUI hand-edits which are no longer available with Rose-Cylc. Ensemble submission is limited in Rose-Cylc. What extra wrapper for Task A will be required to replicate the benefits of Steve's pearl script? BC_RI and OC_RI depend on pcalc files (offline) which may have evolved. How to implement regional emission perturbations - could be perturbed by combining multiple independently perturbed emission inventories, with PPE member index, as with BC_RI and OC_RI.

Submission method (*Grenville*, failed eERC bid). What alternatives will we use for submitting batches of jobs to ARCHER queue?

Space on ARCHER /nerc for temporary holding of completed simulations prior to transfer onto Jasmin for post-processing and long-term storage. Also, our school space is being re-evaluated. I have made long-term use of around 30Tb of space for processing our PPEs and storing output files. We need to decide where to store this data and what alternative will be available for A-CURE.

Early identification of failed jobs due to **poorly behaved parameter perturbation combinations**.

Diagnostics. (Oxford would help identify.) COSP simulators are super-expensive to run. Some diagnostics in key regions (each a diagnostic channel); takes time to set up.

How much 3h data?