SPECIAL PROJECT FINAL REPORT

All the following mandatory information needs to be provided.

Project Title:	Utilising the URANIE platform for sensitivity analysis and optimisation of ensemble perturbation methods in the HARMONIE-AROME model
Computer Project Account:	spiefann
Start Year - End Year :	2024 - 2025
Principal Investigator(s)	James Fannon
Affiliation/Address:	Met Éireann Glasnevin Hill Dublin 9 D09 Y921 Ireland
Other Researchers (Name/Affiliation):	Colm Clancy, Met Éireann Michiel Van Ginderachter, Royal Meteorological Institute of Belgium

The following should cover the entire project duration.

Summary of project objectives

This work focused on utilising the URANIE platform for optimisation of configuration parameters in the Stochastically Perturbed Parameterizations (SPP) scheme in the HARMONIE-AROME NWP model. In particular, a series of tests were carried out to assess the performance of the Efficient Global Optimization (EGO) algorithm, as implemented with URANIE, for finding optimal SPP parameter perturbation standard deviation. The main purpose of this testing was to investigate if URANIE can be used to generalise (and optimise) the manual SPP tuning experimentation which is currently carried out within the HIRLAM and ACCORD consortia. While the original project description also included sensitivity analyses and optimisation of other perturbation schemes in HARMONIE-AROME, due to time constraints this project was ultimately restricted to SPP optimisation only.

Summary of problems encountered

No significant technical problems relating to this Special Project or the HPC facilities at Bologna were encountered.

Experience with the Special Project framework

This was my second Special Project as a Principal Investigator and I again found that all administrative aspects were straightforward. My overall experience with the Special Project framework was very positive.

Summary of results

The computational resources provided by this Special Project enabled significant testing of the URANIE platform for optimisation of SPP configuration parameters in HARMONIE-AROME, with a particular focus on the choice of cost function used in the EGO algorithm and its performance. The main conclusions are summarised below:

- The implementation of the EGO algorithm in HARMONIE-AROME with URANIE works well from a technical perspective.
- With a suitable choice of cost function, the single parameter optimisation workflow can yield quite sensible predictions for an optimal perturbation standard deviation which balances improvement in model performance, as measured by CRPS, against the introduction of systematic biases in the perturbed members relative to the control.
- While initial investigations into multi-parameter optimisation highlighted some of the limitations of the current approach, the overall performance of the EGO scheme was reasonable.
- As such, with some minor adaptions it appears that the general workflow presented in this project could be readily applied to wider SPP configuration tuning with HARMONIE-AROME and potentially provide some benefit over current tuning strategies.

For a complete analysis and discussion of the results please see the detailed internal technical report appended to this document.

List of publications/reports from the project with complete references

Fannon J (2024). Utilising the URANIE platform for optimisation of the stochastically perturbed parameterizations scheme in HARMONIE-AROME. Met Éireann NWP Note 2025/01. [Internal technical report, appended to this report]

Future plans

This work will form the basis for further mutli-parameter SPP optimisation testing in the near future, the results of which will be compared against existing operational SPP configurations. The general URANIE framework outlined in this project will also be extended to the single-column version of HARMONIE-AROME, MUSC, in 2025 as part of an ACCORD-funded working week.

Met Éireann
NWP Note 2025/01

ECMWF Special Project 2024 Report

Utilising the URANIE platform for optimisation of the stochastically perturbed parameterizations scheme in HARMONIE-AROME

James Fannon

With thanks to Michiel Van Ginderachter (RMI)

Met Éireann
2025

Abstract

This NWP note provides an overview of work carried out as part of an ECMWF Special Project, SPIEFANN, in 2024. This work focused on utilising the URANIE platform for optimisation of configuration parameters in the Stochastically Perturbed Parameterizations (SPP) scheme in the HARMONIE-AROME model. In particular, a series of tests were carried out to assess the performance of the Efficient Global Optimization (EGO) algorithm, as implemented with URANIE, for finding optimal SPP parameter perturbation standard deviation. The main purpose of this testing was to investigate if URANIE can be used to generalise (and optimise) the manual SPP tuning experimentation which is currently carried out within the HIRLAM and ACCORD consortia.

The results presented herein suggest that:

- The implementation of the EGO algorithm in HARMONIE-AROME with URANIE works well from a technical perspective.
- With a suitable choice of cost function, the single parameter optimisation workflow can yield quite sensible predictions for an optimal perturbation standard deviation which balances improvement in model performance, as measured by CRPS, against the introduction of systematic biases in the perturbed members relative to the control.
- While initial investigations into multi-parameter optimisation highlighted some of the limitations of the current approach, the overall performance of the EGO scheme was reasonable.

As such, with some minor adaptions it appears that the general workflow presented in this note could be readily applied to wider SPP configuration tuning with HARMONIE-AROME and potentially provide some benefit over current tuning strategies.

Contents

1 Introduction

URANIE is a sensitivity and uncertainty analysis platform developed at CEA (the French Alternative Energies and Atomic Energy Commission) and based on the ROOT framework (developed at CERN, see `https://root.cern/`). URANIE contains a wide variety of tools for uncertainly propagation, sensitivity analysis, optimisation problems, and more. The code is open-source and available at `https://sourceforge.net/projects/uranie/` (see associated documentation therein).

Initial work on integrating URANIE within the HARMONIE-AROME workflow was carried out by Michiel Van Ginderachter (RMI) as part of the ESCAPE-2 project (Van Ginderachter, 2022). Several proof of concept experiments were carried out to illustrate how URANIE's sensitivity and optimisation tools could be used to tackle different problems within HarmonEPS (i.e. the ensemble realisation of the HARMONIE-AROME model). In particular, these experiments focused on:

1. a Morris Screening sensitivity analysis of the surface perturbation scheme to investigate the dry bias of ensemble members in HARMONIE-AROME cycle 40h.1.1, and
2. tuning of the SPP scheme correlation length scale using the EGO algorithm (see Section 2.3.2).

Further details regarding thesee experiments can be found in Van Ginderachter (2021).

URANIE was first explored in Met Éireann in 2023. Based on the work discussed above, URANIE was implemented within HARMONIE-AROME cycle 46 on ECMWF's ATOS HPC platform and various HarmonEPS sensitivity analyses were carried out (both of the surface perturbation and SPP schemes). Initial investigations into SPP optimisation using the EGO algorithm took place as part of an ACCORD scientific visit in late 2023 (Van Ginderachter and Fannon, 2024). This visit highlighted the crucial role played by the cost function used during optimisation and emphasised the need for a more rigorous assessment of overall performance.

This NWP note describes significant testing of the URANIE platform for optimisation of SPP configuration parameters in HARMONIE-AROME, with a particular focus on the choice of cost function used for optimisation and the performance of the EGO algorithm. The computational resources used for this testing was provided by an ECMWF Special Project in 2024 (SPIEFANN). One can note that the original project description also included sensitivity analyses and optimisation of other perturbation schemes in HarmonEPS (e.g. the surface perturbation scheme, Fannon and Clancy (2024)). However due to time constraints the project was ultimately restricted to SPP optimisation only.

The remainder of this note is structured as follows. Section 2 provides a description of the HARMONIE-AROME configuration settings and the overall strategy for the SPP optimisation experiments. A brief overview of URANIE is given, focusing solely on installation, incorporation into the HARMONIE-AROME workflow, and the EGO algorithm, along with a brief sidenote on SPP pattern reproducibility in HARMONIE-AROME cycle 46. In Section 3 optimisation of a single SPP configuration parameter is considered for several SPP parameters, while Section 4 is a brief investigation into extending the optimisation problem to two SPP configuration parameters. The note concludes in Section 5 with an overview of results and outlook for future work.

2 Technical details

2.1 HARMONIE-AROME configuration settings

The dev-CY46h1_eps branch of HARMONIE-AROME was used for all experiments discussed in this NWP note. A frozen version of this branch[1], which was the latest available at the start of the project, was used throughout. It can be noted that this frozen version of the branch lags significantly behind the latest version of the branch at the time of writing, and this should be borne in mind when interpreting results presented herein. Note also that the tagged version of cycle 46 (i.e. harmonie-46h1.1) was not available at the start of this project.

General configuration settings are given in Table 1, where all other namelist and configuration settings can be assumed to be the default used in the dev-CY46h1_eps branch at that time (unless otherwise stated). A parent-child approach is taken for the ensemble experiments, as summarised below:

1. A "parent" experiment is first run for the control member only over the period of interest. This is a standard 3-hour data assimilation cycling experiment with long runs (36-hour) at 00 UTC. This parent experiment therefore produces analysis files for each forecast start date.

2. A "child" experiment is then run for the perturbed ensemble members. Six members are used here following standard practice in ACCORD. For a given forecast start date, each perturbed member starts from the control member analysis files (i.e. the upper-air MXMIN1999+0000 and surface ICMSHANAL+0000.sfx files) produced by the parent experiment.

Component	**Parent experiment**	**Child experiment**
Precision	dual	-
Domain	IRELAND25S, L65, QUADRATIC grid	-
Data Assimilation	3DVAR, CANARI_OI_MAIN (MARS conv. only)	No DA
LSMIXBC	yes	no
Boundaries	IFSHRES (UWC-W archive)	-
ENSMSEL	0	1-6
ENSCTL_IS_PRESENT	yes	no
Perturbations	None	SPP only
Cycles	+36h at 00 UTC, 3h cycling otherwise	+36h at 00 UTC
Compiler	gnu	-

Table 1: Configuration settings common to all parent and child experiments, unless otherwise stated. Missing entries in the child column are assumed the same as the parent. Note that "No DA" refers to ANAATMO=ANASURF=none and "SPP only" refers to SPP=yes, PERTATMO=PERTSURF=ENSINIPERT=none, and SLAF commented out.

This approach has a number of benefits:

1. The control member only has to be run once, instead of repeating it for each ensemble experiment.

[1] At commit 6730819, see `https://github.com/Hirlam/Harmonie/commits/6730819886a4de9d6a6c2a2f822a940727b81e0d/`.

2. The data assimilation cycles for the perturbed members are skipped, which speeds up experiment runtime considerably.

3. Starting all perturbed members from the same point allows for a direct assessment of the impact of individual perturbations as they are isolated from all other external factors.

As discussed in Section 1, only SPP perturbations will be considered in the ensemble experiments.

Note that the domain IRELAND25S differs slightly from the IRELAND25 domain generally used in technical testing (see Figure 1). A square domain was used here in order to ensure reproducibility of the SPP perturbation patterns (see Section 2.5). All IRELAND25S experiments with data assimilation switched on utilised structure functions from the IRELAND25_090 domain[2], which therefore avoided generating new structure functions. Some additional details regarding the input data for the experiments can be found in Appendix 6.1.

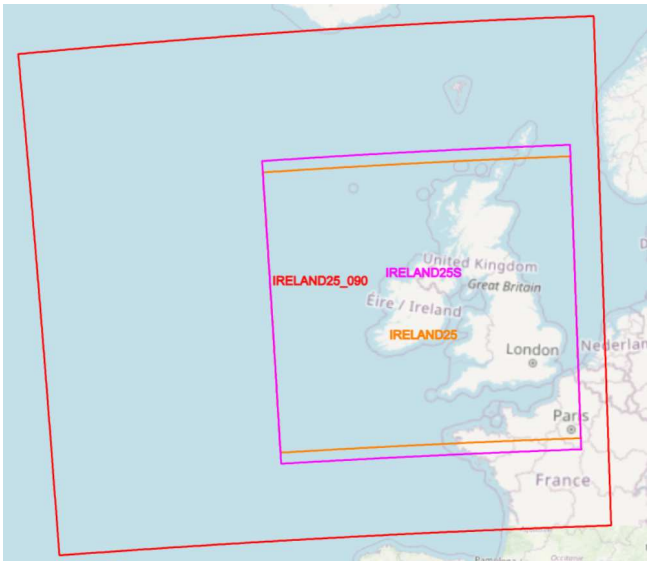


Figure 1: The square IRELAND25S domain (magenta, 540x540 points) used in this NWP note, along with the standard IRELAND25 (orange) and IRELAND25_090 (red) domains.

2.2 SPP and tuning

The SPP scheme in HARMONIE-AROME introduces stochastic perturbations to selected closure parameters in the physical parameterizations of the model, such as the microphysics and turbulence schemes (Frogner et al., 2022), and to the model dynamics[3]. There are currently sixteen SPP parameters available in HARMONIE-AROME; the fifteen described in Tsiringakis et al. (2024) and the recently introduced "SLWIND" dynamics perturbation. SPP has been used operationally at ECMWF (Lang et al., 2021) and MetCoOp for several years, and is also used operationally in the UWC-West consortium since March 2024.

In HARMONIE-AROME SPP perturbations for a given parameter are generated using the Stochastic Pattern Generator (SPG) routine, with perturbation patterns evolving in space and time according to

[2]Which is possible as IRELAND25S lies within IRELAND25_090 (see Figure 1) and the two domains used the same horizontal and vertical resolution.

[3]In the semi-lagrangian advection scheme.

specified spatial and temporal correlation scales (see Tsyrulnikov and Gayfulin (2017) for details). The perturbation patterns are typically updated every hour and linearly interpolated at intermediate timesteps, however the pattern update frequency can be modified. Some SPG settings, which are common to all experiments discussed, are indicated in Table 2.

The perturbed parameter values are drawn from either lognormal or pseudo-uniform distributions which are typically (but not always) centered around the default value used for the parameter in the model (i.e. that used by the control member). For a lognormal distribution, the SPP perturbation pattern, P, for a given SPP parameter, p, is described by

$$P = d_p \times \exp\left(\mu_p + \sigma_p \times I_p \times R_p\right), \quad \mu_p = \begin{cases} -\left(\sigma_p \times s\right)^2/2 & \text{if } m_p \text{ is TRUE,} \\ 0 & \text{if } m_p \text{ is FALSE,} \end{cases} \tag{1}$$

while for a uniform distribution

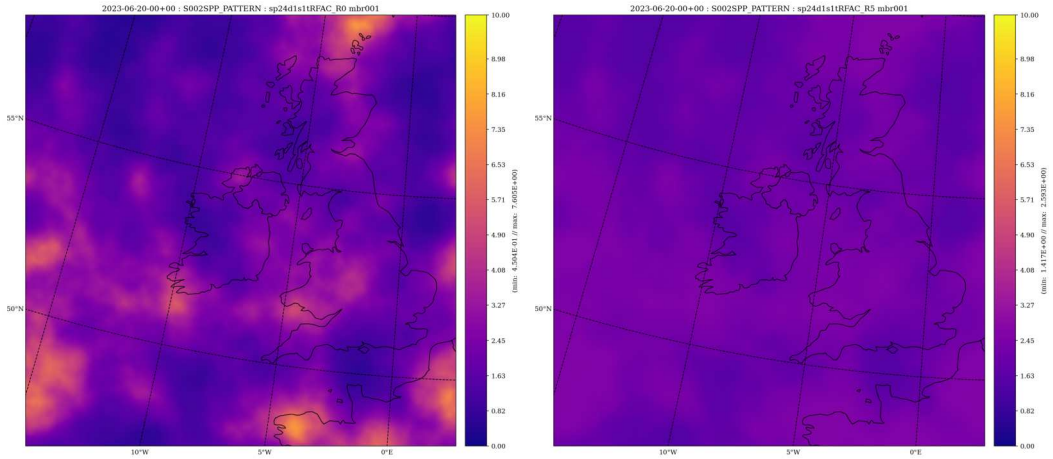
$$P = d_p + z_p \times \sigma_p \times \left(\frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{I_p \times R_p}{\sqrt{2} \times s}\right)\right] - o_p\right), \quad z_p = \begin{cases} 1 & \text{if } d_p = 0, \\ d_p & \text{otherwise,} \end{cases} \tag{2}$$

where R_p represents the random field generated by SPG and erf is the error function. All other variables are described in Table 2 along with their corresponding namelist entry in harmonie_namelists.pm and other relevant SPP settings. Note that clipping of the perturbed field P at lower and upper bounds can also be specified.

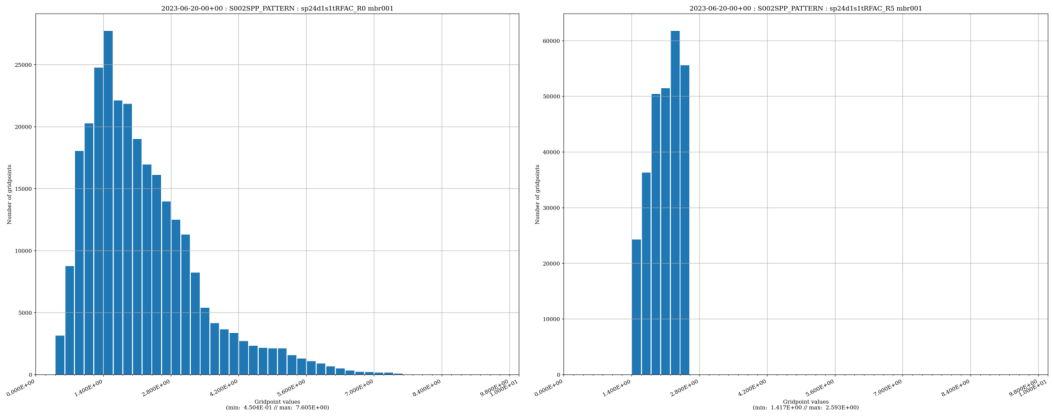
Variable	**Description**	**Namelist entry**	**Value**
TAU_SPP	SPG temporal correlation scale	TAU	43,200 s
XLCOR_SPP	SPG spatial correlation scale	XLCOR	200 km
NPATFR_SPP	SPG pattern update frequency	NPATFR	-1 (every hour)
d_p	Default value for p	-	Variable
σ_p	Perturbation standard deviation for p	CMPERT_p	Variable
I_p	Correlation for p	-	1
s	Pattern standard deviation	SDEV	1
m_p	Mean/median lognormal distribution	LLNN_MEAN1_p	TRUE
o_p	Offset for p (uniform only)	UNIFORM_OFFSET_p	Variable
LPERT_p	Activate SPP for p	LPERT_p	TRUE/FALSE
LUNIFORM_p	Uniform distribution for p	LUNIFORM_p	TRUE/FALSE
CLIP_p	Clipping of P	CLIP_p	Variable

Table 2: SPP-relevant variables and values used in this note (if applicable). The column "Namelist entry" refers to how the variable appears in harmonie_namelists.pm and p refers to the SPP parameter. Note that I_p is set in arpifs/setup/get_spp_conf.F90.

Sample SPP perturbation patterns for the the SPP parameter "RFAC_TWOC" are illustrated in Figure 2. As m_p is TRUE in this case the mean value of the lognormal distribution is approximately equal to the default value of 2 for the model parameter (i.e. RFAC_TWOC_COEF). This is also the case for the uniform distribution as the offset $o_p = 0.5$ ensures that the perturbations are centered around the default. One can note that for a fixed value of σ_p, i.e. the perturbation standard deviation, the lognormal distribution gives perturbations which deviate more significantly from the default value.



(a) Fields



(b) Histograms

Figure 2: Sample SPP perturbation patterns for RFAC_TWOC. A lognormal distribution with $d_p = 2$ and $\sigma_p = 0.6$ is illustrated on the left, while the corresponding uniform distribution with $o_p = 0.5$ is given on the right. A clipping of $0 - 10$ is applied and the SPG field, R_p, is common to both.

Tuning of the SPP scheme for a given cycle of HARMONIE-AROME is typically first done on an individual parameter-by-parameter bias, where the impact of different perturbation distribution settings (e.g. lognormal or uniform, standard deviation, and offset) on ensemble performance is assessed. Assessment is generally done via point verification metrics such as CPRS, spread-skill, and ensemble member bias relative to the control. Groups of SPP parameters are then tested in combination to assess overall performance, and this process ultimately leads to recommended SPP configuration settings for operational use. For example, both MetCoOp and UWC-W use a five parameter SPP configuration. See Frogner et al. (2022) and Tsiringakis et al. (2024) for further details.

SPP tuning can be a somewhat laborious and time-intensive process. It is also clear that the large number of degrees of freedom makes arriving at settings for an optimal multi-parameter SPP configuration very challenging. As such, it would be highly desirable if the optimisation routines available in URANIE could be used to help to:

- automate the process of SPP tuning,
- guide the selection of SPP settings for both individual SPP parameters and multi-parameter SPP configurations.

The framework for doing this with the EGO algorithm will be outlined in Section 2.3.2.

2.3 URANIE

URANIE is a powerful and complex platform with a significant learning curve. The main focus of this project was to utilize aspects of URANIE to guide ensemble development in HARMONIE-AROME, and as such an in-depth overview of the platform is beyond the scope of this NWP note. The scripting of the EGO algorithm with URANIE, as described in Section 2.3.2, was written by Michiel Van Ginderachter (RMI) and will largely by used as a black-box for our purposes. Installation instructions on ATOS are also thanks to Michiel.

2.3.1 Installation on ATOS

A pre-compiled version of URANIE version 4.8.0, made available by Michiel, was used throughout this project to ensure consistency. This can be used by adding

```
source /hpcperm/cu0k/URANIE/uranie.env
```

in your shell script before calling URANIE. For reference, installation instructions for URANIE on the ATOS HPC are given below:

1. Download the URANIE source code and additional libraries which are not available on ATOS as modules:
   (a) URANIE: `https://sourceforge.net/projects/uranie/`
   (b) ROOT (v6.28.04): `https://github.com/root-project/root/tree/v6-28-04`
   (c) cppunit (v1.15.1): `https://github.com/MITK/CppUnit/tree/cppunit-1.15.1`
   (d) NLopt (v2.6.1): `https://github.com/stevengj/nlopt/tree/v2.6.1`
2. Set your environment by loading the required modules:

```
module load prgenv/gnu
module load gcc/8.5.0
module load cmake/3.19.5
module load fftw/3.3.8
module load openmpi/4.1.1.1
module load doxygen
module load python3/3.10.10-01
```

3. Install ROOT, cppunit, and NLopt by following the instructions in the URANIE README.
4. Once these are installed, set the following environment variables and install URANIE following the README.

```
ROOTSYS=<root_installation_dir>
CPPUNITSYS=<cppunit_installation_dir>
NLOPTSYS=<nlopt_installation_dir>
export PATH=${NLOPTSYS}/bin:${CPPUNITSYS}/bin:${ROOTSYS}/bin:$PATH
export LD_LIBRARY_PATH=${NLOPTSYS}/lib:${CPPUNITSYS}/lib:${ROOTSYS}/lib:$LD_LIBRARY_PATH
```

2.3.2 EGO algorithm

An overview of the EGO algorithm in the context of NWP tuning and optimisation is given below. A more general description can be found in Jones et al. (1998).

Suppose one wishes to study the impact of certain NWP model parameters on a particular aspect of model performance. For example, one may may be interested in assessing how different closure parameters in a physical parametrization scheme influences the RMSE of 2 m temperature at a leadtime of 24 hours, for a given model cycle, domain, period etc.. Let $X = x_1, ..., x_n$ denote the input parameters of interest and F denote the cost function (i.e. 2 m temperature RMSE). The EGO algorithm will attempt to find the optimal values of the array X such that the cost function F is minimised. The algorithm proceeds as follows:

1. Take N_T initial samples of the input parameter space X.
2. Evaluate the cost function F for each parameter space sample. For NWP, this entails running the model for each sample, i.e. with parameter values $X^j = x_1^j, ..., x_n^j$, and performing the verification to produce the cost function response F^j.
3. Train a Kriging model using all of the X^j and F^j available. This surrogate model attempts to find a function G such that $F = G(X)$.
4. Find the maximum Expected Improvement (EI) based on kriging variance to generate a new parameter sample Y. Check if this new sample meets some user-specified stopping criteria.
5. If the stopping criteria are not met, then re-evaluate the cost function F for the new Y and repeat steps 3-5. Continue this process until a maximum number of iterations ($N_{\max}$) is reached.

Implementation of the EGO algorithm with URANIE and its use for SPP tuning in HARMONIE-AROME was originally developed in Van Ginderachter (2021). The initial proof-of-concept experiment considered a single input parameter, the SPP spatial correlation length scale XLCOR_SPP, to optimise, while the target cost function was 2 m temperature CRPS. We follow an analogous approach in this work, where the input parameters to optimise will be associated with SPP perturbation distribution settings, and in particular the perturbation standard deviation σ_p (also typically referred to as CMPERT, see Table 2). While there are of course significant limitations associated with point verification metrics, the target cost functions used herein will be derived from such verification scores. This is a pragmatic choice which mirrors what is typically used in the manual SPP tuning described in Section 2.2.

Some technical information regarding URANIE-specific settings for the EGO algorithm are provided in Appendix 6.2. Further details regarding EGO in URANIE can be found in the URANIE user manual (see `https://sourceforge.net/projects/uranie/`).

2.3.3 Incorporating in HARMONIE-AROME

The basic strategy for incorporating URANIE into HARMONIE-AROME is to add additional ecflow jobs in the HARMONIE-AROME scripting system to carry out individual URANIE tasks, a typical example of which is illustrated in Figure 3. The workflow proceeds as follows, with specific reference to steps in the EGO algorithm.

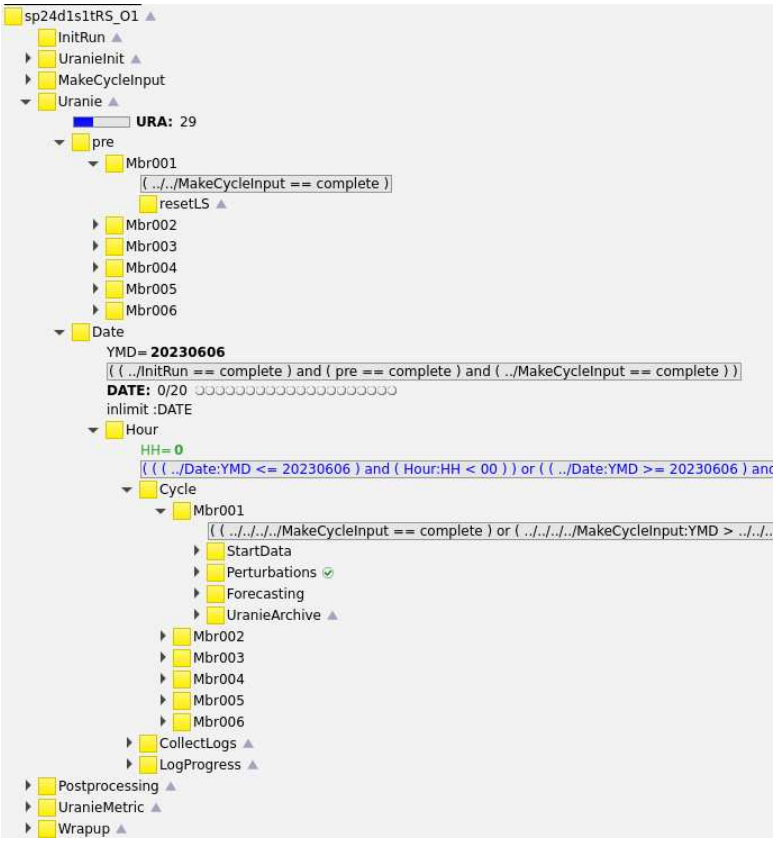


Figure 3: A typical HARMONIE-AROME experiment with URANIE activated.

UranieInit

This task initializes URANIE and generates the N_T initial samples of the input parameter space to be used in the experiment. A so-called "design-of-experiments" file is generated and placed in $HM_DATA/URANIE/init_doe.dat which contains the N_T samples. Also included in this directory are UranieLauncher_i directories, where $i \in [1, N_T]$, which includes data specific to each individual parameter sample i.

In the context of EGO for SPP tuning, the "UranieInit" task represents step 1 of the algorithm. As our main interest here will be optimizing the SPP perturbation standard deviation σ_p (see Section 2.3.2), the samples generated will typically represent different values for σ_p limited between some user-specified minimum and maximum values. A sample init_doe.dat file for SLWIND σ_p and $N_T = 10$ may look like:

```
#COLUMN_NAMES: SLWIND_CMPERT| tds__n__iter__
#COLUMN_TYPES: D|D

2.856639807e-01 1
1.940128760e-01 2
1.553383534e-01 3
2.577281340e-01 4
7.340553306e-02 5
2.175854434e-01 6
3.794922704e-02 7
3.893414641e-01 8
3.306176611e-01 9
1.190635072e-01 10
```

For each sample i the "UranieInit" task will then generate a harmonie_namelists.pm file where the value of SLWIND σ_p is replaced by the sample value. This harmonie_namelists.pm file is placed in the $HM_DATA/URANIE/UranieLauncher_i directory. For example:

```
$ grep CMPERT_SLWIND UranieLauncher_1/harmonie_namelists.pm
'CMPERT_SLWIND' => 2.856639806903e-01 ,
```

MakeCycleInput

The standard HARMONIE-AROME task for generating LBCs. In the context of EGO for SPP tuning, all boundaries are generated before the "Uranie" job as the boundaries are shared across the entire URANIE experiment.

Uranie

This wraps the standard "Date" family in HARMONIE-AROME and includes an additional "pre" step for any URANIE specific data processing before the forecast is run. The "URA" counter indicates the current sample i or "iteration" under consideration, ranging from $[1, N_{\max}]$ where $N_{\max}$ is specified by the user. Henceforth we use the term "iteration" to refer to this counter. In the context of EGO for SPP tuning, for iteration i the "resetLS" task copies over the harmonie_namelists.pm file from $HM_DATA/URANIE/UranieLauncher_i to the experiment $HM_LIB. As such, the forecast corresponding to iteration i will use the sampled values for σ_p etc. contained therein.

The "Date" family then runs as in a standard HARMONIE-AROME experiment, running over all forecast start dates. An extra task, "UranieArchive", is included after the "Forecasting" task to archive forecast data specific to iteration i (e.g. ICM files) in $HM_DATA/URANIE/UranieLauncher_i if desired. This is required as each iteration overwrites data from the previous iteration. In the context of EGO for SPP tuning, the "Uranie" task represents part of step 2 of the algorithm.

Postprocessing

The standard HARMONIE-AROME task for post-processing which triggers once "Uranie" is complete.

UranieMetric

This task archives the verification data for iteration i into $HM_DATA/URANIE/UranieLauncher_i and computes the cost function for this iteration if required. In the context of EGO for SPP tuning, this tasks represents part of step 2 of the algorithm. The following is then carried out based on the current iteration i:

- If $i < N_T$, move on to the next iteration by requeueing the "Uranie" and "Postprocessing" families.
- If $N_T \leq i < N_{\max}$, run the EGO algorithm using all available input and cost function samples to generate a new sample Y and check if it meets the stopping criteria (i.e. step 3 and 4 of the

algorithm). If Y does not meet the criteria then move on to the next iteration, which will use sample Y, by requeueing the "Uranie" and "Postprocessing" families. This represents step 5 of the algorithm.

- If $i = N_{\max}$ the maximum number of iterations has been reached and the experiment is set complete.

Once the experiment with URANIE has completed, the parameter samples used and associated cost function values will be available in $HM_DATA/URANIE/temp.dat, e.g.

```
#COLUMN_NAMES: SLWIND_CMPERT| cf_crpsmb| tds__n__iter__
#COLUMN_TYPES: D|D|D

2.856639807e-01 4.307089000e+00 1
1.940128760e-01 4.727790000e-01 2
1.553383534e-01 4.883700000e-02 3
2.577281340e-01 2.743437000e+00 4
7.340553306e-02 2.910070000e-01 5
2.175854434e-01 1.674062000e+00 6
3.794922704e-02 6.075070000e-01 7
3.893414641e-01 1.280035900e+01 8
3.306176611e-01 8.200705000e+00 9
1.190635072e-01 6.187600000e-02 10
1.377236073e-01 1.143720000e-01 11
1.685138327e-01 1.851800000e-01 12
1.029639329e-01 8.769850000e-01 13
1.000000000e-02 1.523795000e+00 14
1.460431296e-01 3.009830000e-01 15
1.291546573e-01 5.612100000e-02 16
1.282993470e-01 2.060320000e-01 17
5.891114732e-02 2.082420000e-01 18
1.762451172e-01 1.519270000e-01 19
1.603729636e-01 2.355780000e-01 20
1.238940371e-01 3.347620000e-01 21
1.348671620e-01 5.268400000e-02 22
1.346767323e-01 3.082010000e-01 23
1.787970358e-01 6.636520000e-01 24
1.518701172e-01 4.863620000e-01 25
6.067138090e-02 1.134620000e-01 26
5.727349730e-02 1.302150000e-01 27
6.401554172e-02 1.928530000e-01 28
5.475561039e-02 2.054470000e-01 29
```

where the rightmost column represents the URANIE iteration, the second from right column represents the cost function value, and the other columns are the samples for each parameter.

All of the technical changes required to incorporate URANIE into HARMONIE-AROME are included in the following repository:

`https://github.com/mpvginde/UranEPS/tree/feature/spiefann24_setup`

Note that this repository is intended to be built upon the dev-CY46h1_eps branch of HARMONIE-AROME used in this project, and thus only contains files which are changed relative to this.

2.4 SBU estimates

As suggested by the workflow in Section 2.3.3, utilizing URANIE in HARMONIE-AROME can quickly become computationally demanding. Take for example a typical child experiment with the HARMONIE-AROME configuration indicated in Section 2.1 and EGO activated, i.e. six ensemble members, a 36-hour forecast, and one forecast cycle per day. A single 36-hour forecast in single

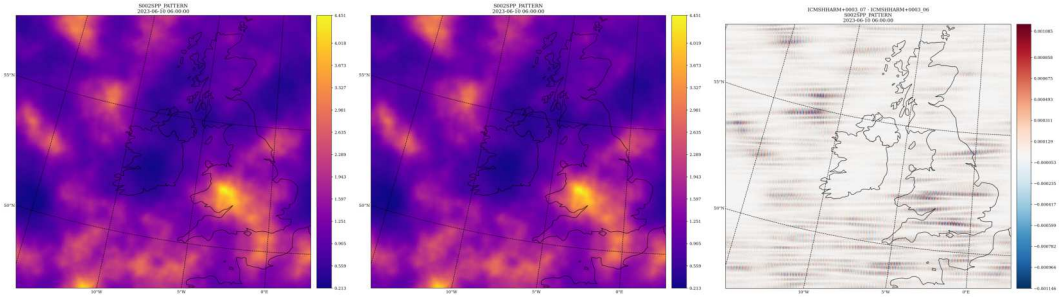
precision over the IRELAND25S domain costs approximately 3,000 SBUs. As such, an estimate for the cost of the forecasts associated with a single EGO HARMONIE-AROME experiment is:

$$\begin{aligned} C = D(\text{days}) \times 1(\text{cycles/day}) \times 6(\text{members}) \times N(\text{iterations}) \times 3,000(\text{single forecast SBU}) \\ \implies C = 18,000 \times D \times N \end{aligned} \tag{3}$$

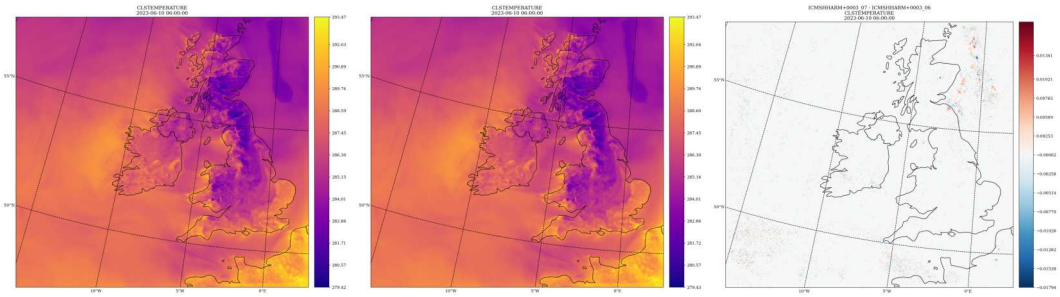
where N refers to the number of URANIE iterations. Assuming a rough estimate of $N_{\max} = 50$, then a single cycle/two week experiment can cost up to 900 K/12.6 M SBU, respectively. As such running two week periods for verification, which is standard practice in ACCORD, can be quite expensive. Therefore much of the technical testing in this project used only a single cycle, while the longer optimisation tests used a one week period.

2.5 Sidenote on SPP pattern reproducibility

As part of initial technical testing with the IRELAND25 domain it was noted that the SPP patterns generated using the HARMONIE-AROME version used were not reproducible i.e. the same experiment run twice gave slightly different SPP perturbation patterns, see Figure 4. While the pattern differences are clearly small, they can result in not insignificant forecast differences between two identical experiments at longer leadtimes, which of course will result in different verification output. While such non bit-reproducible results would not generally be of concern for ensemble experiments, in this work we will frequently compare the cost function for a given URANIE iteration to the cost function from a reference experiment (e.g. see Section 3.1.3). Therefore having a non-reproducible cost function for the reference experiment will significantly undermine this comparison.



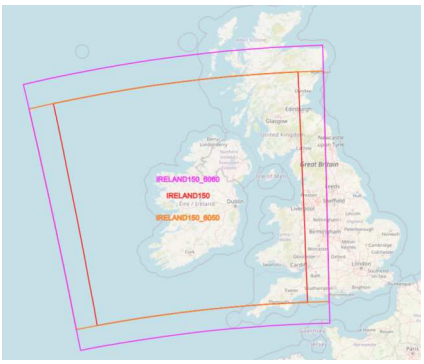
(a) S002SPP_PATTERN



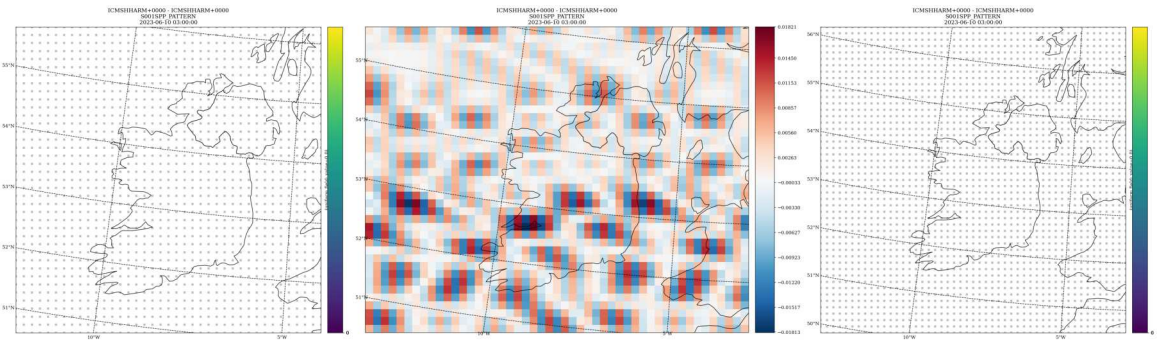
(b) CLSTEMPERATURE

Figure 4: Sample (a) SPP perturbation and (b) 2 m temperature fields from two identical experiments (left and middle columns) using the IRELAND25 domain (see Table 1 for other settings). The right column is the difference of the left and middle columns.

After further technical tests it was found that the SPP patterns were reproducible when using a square domain. This is illustrated in Figure 5 using several 15 km domains over Ireland; IRELAND150_6050 is an extension of the square IRELAND150 domain by 10 points in east-west, while IRELAND150_6060 (magenta) is an extension of IRELAND150 by 10 points in east-west and north-south. For the square domains, the SPP perturbation patterns are reproducible from run-to-run, whereas this is not the case for the non-square IRELAND150_6050 domain. The need for reproducible patterns therefore motivated the choice of the IRELAND25S domain (see Figure 1) in this project.[4]



(a) Test domains



(b) SPP pattern differences

Figure 5: (a) Sample 15 km test domains IRELAND150 (red), IRELAND150_6050 (orange), and IRELAND150_6060 (magenta). See text for details. (b) Sample SPP pattern differences to test for pattern reproducibility over domains IRELAND150 (left), IRELAND150_6050 (middle), and IRELAND150_6060 (right).

[4]One can note that upon further analysis it was found that switching from the gnu to intel compiler resolved this issue for all domains.

3 Single parameter optimisation

Based on experience from a 2023 ACCORD scientific visit on URANIE, the approach taken in this project was to use relatively simple optimisation problems (i.e. with one or two parameters) and assess the performance of the EGO scheme for parameter tuning with different cost functions. As such, in this section we present results for EGO of a single parameter in HARMONIE-AROME; namely the SPP perturbation standard deviation σ_p for a single SPP parameter. In Section 3.1 the SPP parameter of interest will be "RFAC_TWOC" (i.e. perturbations of the top entrainment term "RFAC_TWO_COEF" in the model) and will deal moreso with technical testing. The "SLWIND" SPP parameter (i.e. perturbations of V(M) in the semi-lagrangian advection scheme) will be considered in Section 3.2 along with more realistic optimisation tests.

3.1 RFAC_TWOC

Previous SPP tuning results illustrated that σ_p for RFAC_TWOC, henceforth denoted as σ_R, was found to have a significant impact on cloud cover and cloud base verification scores, particularly during the summer (not shown). As such, based on advice from EPS experts, RFAC_TWOC was chosen as a suitable SPP parameter to focus on for initial optimisation testing.

3.1.1 Experiment details and verification

All experiments followed the configuration described in Section 2.1. A two week summer period (2023/06/06/00 - 2023/06/20/00) was considered with a five day spin-up for the control member. In order to reduce the costs associated with EGO experiments (see Section 2.4), only a single forecast start date (2023/06/06/00) was considered for much of the technical testing.

The only perturbations active in the ensemble are SPP. As such, given that a single SPP parameter is considered here, the only perturbation active is the RFAC_TWOC perturbation. Therefore in the case of $\sigma_R = 0$ all perturbed members collapse to the control member (see Appendix 6.3). SPP settings are as described in Table 2 with the following:

- $d_p = 2$ (default value for RFAC_TWOC_COEF),
- $\sigma_p = \sigma_R$ is the single variable to be optimised using EGO,
- A uniform distribution with $o_p = 0.5$ (unless otherwise stated, see Section. 3.1.2)
- perturbation clipping set to 0-10.

All cost functions for the EGO algorithm considered below will be based on point verification metrics. While such metrics have their drawbacks, this approach largely mirrors what is currently done within HIRLAM and ACCORD for SPP tuning. Verification scores were generated using the harp package (`https://github.com/harphub/`), and in particular the oper-harp-verif scripts (`https://github.com/harphub/oper-harp-verif`). Verification is carried out over all available synoptic stations in the IRELAND25S domain with local Met Éireann vobs files utilised for all variables considered.[5]

[5]Note that the "error_sd" and "fctmax_val" observation screening tools in the oper-harp-verif scripts were not used for all parameters. This was to ensure that the same set of observations was used in each experiment.

3.1.2 Reference runs and cost function analysis

In order to first gauge the impact of σ_R on model performance a number of reference experiments were carried out over the two week period. These reference experiments differ only in the value of σ_R used, which was manually chosen for each experiment. To help assess a sensible range of values for the perturbation standard deviation, one can consider the SPP perturbation distributions for RFAC_TWOC given by (1) and (2), as illustrated in Figure 6.

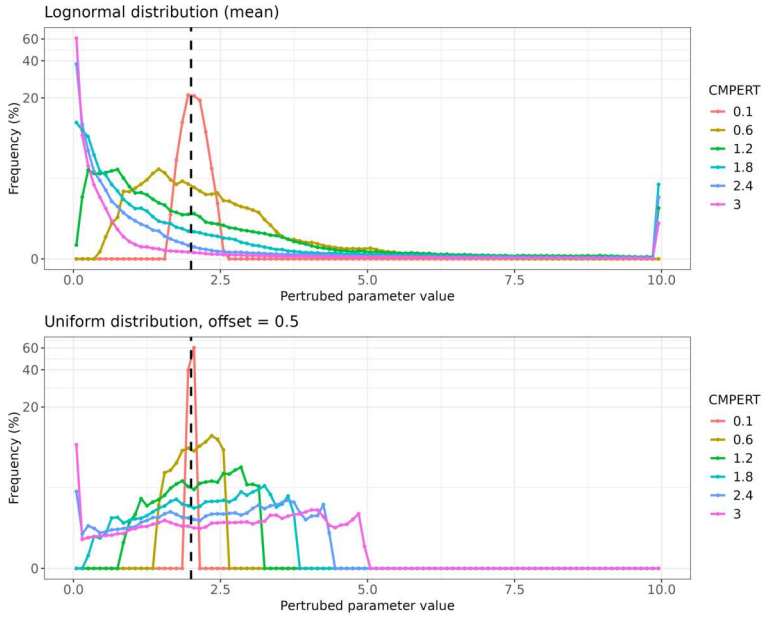


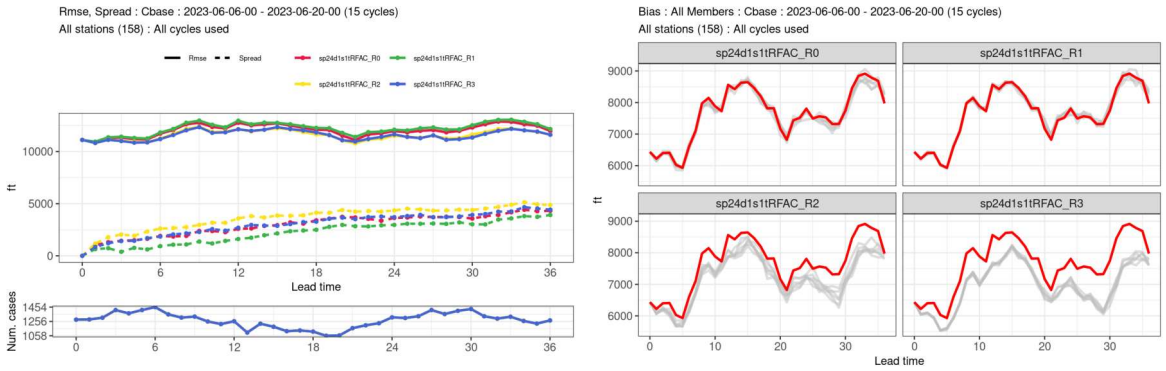
Figure 6: SPP perturbation distributions for RFAC_TWOC according to (1) (top) and (2) (bottom). Parameter values are indicated in Table 2 with $d_p = 2$, $o_p = 0.5$, and clipping between 0-10. The colors correspond to different values for σ_R, while the R_p used is a sample taken from a HARMONIE-AROME experiment.

At large values of σ_R the perturbation distributions become quite unrealistic due to the high frequency of values at the clipping extremes and the prevalence of values outside the recommended range[6] for RFAC_TWOC (i.e. $\approx 0.5 - 3$). This is particularly evident for the lognormal distribution, where σ_R should be limited to approximately 1.2 to avoid producing a somewhat binary distribution with values of 0 and 10 in this case. Figure 7 demonstrates that the model spread actually decreases at extreme σ_R due to this effect, with a strong systematic bias in the perturbed members relative to the control also evident. Lognormal distributions for the SPP perturbations tend to introduce such biases in the pertubed members, with Tsiringakis et al. (2024) recommending the use of uniform distributions to alleviate this problem. As such uniform distributions will be used henceforth, which also typically allows for a larger σ_p range to be considered. Based on Figure 6, a range of $\sigma_R \in [0.1, 2.4]$ was deemed suitable for exploration.

The impact of σ_R on total cloud cover and cloud base is illustrated in Figure 8 for some of the standard EPS verification metrics (i.e. spread-skill, CPRS, and member bias).[7] It is important to emphasise again that only the RFAC_TWOC SPP perturbation is switched on, a uniform distribution with $o_p = 0.5$ is used, and σ_R is limited to values between 0.1-2.4. One can observe a relatively

[6]See recommended ranges at `https://hirlam.github.io/HarmonieSystemDocumentation/dev/EPS/SPP/`

[7]These cloud parameters were considered based on advice from EPS experts.



(a) Spread-skill

(b) Member bias

Figure 7: Cloud base point verification over the two-week summer period with a lognormal distribution of RFAC_TWOC and varying σ_R. (a) Spread-skill for $\sigma_R = 0.6$ (red), 0.1 (green), 2.4 (yellow), and 6 (blue) and (b) corresponding member bias (reading row wise from top-left to bottom-right).

linear response in the model spread and CRPS to σ_R, while the impact on ensemble mean RMSE is quite weak. At larger values of σ_R (1.8 and 2.4) the pertubed members become slightly biased with respect to the control member (positively for total cloud cover, negatively for cloud base) at longer lead times. Note that as the perturbed members all start from the control analysis for each forecast start date, a perturbed member bias (if any) relative to the control will typically become most evident towards the end of the forecast. If each ensemble member was cycling independently, it's possible that such member biases would also be evident at the start of the forecast.

In the context of choosing a suitable cost function (denoted as F henceforth) for optimisation of σ_R using EGO, the standard verification scores in Figure 8 represent some obvious candidates. There are clearly a number of choices which must be made when designing F based on such metrics, e.g.

- What verification score (e.g. CRPS, CRPS potential/reliability, spread-skill ratio (SSR)) or combination of scores should be used?
- What model parameter (e.g. 2 m temperature, 10 m wind speed) or combination of parameters should be used?
- How should the selected score(s) for the chosen parameter(s) be condensed down to a single value for the cost function e.g. take CRPS at 24-hours, average CRPS over all lead times, etc.?

As a starting point it is reasonable to first focus on using a single model parameter (i.e. total cloud cover or cloud base) when formulating F. After some initial evaluation, it was found that using verification scores at a given lead time can mask more systematic trends in the scores and thus be quite noisy, while averaging a score over the entire forecast lead time is impacted by the fact that the perturbed members start from the control (thus reducing the difference between experiments). As such averaging scores over the last 24-hours of the forecast (i.e. hours 12-36 in this case) was deemed a more appropriate choice when computing F as it focuses on a period where the model perturbations have had time to develop. This lead time averaging will be assumed henceforth.

In Figure 9 we plot the variation in possible choices for F as a function of σ_R for the reference experiments in Figure 8. Here we first consider metrics which are direct output from harp (i.e. CRPS, RMSE, and SSR) and the Mean Member Bias Relative to the Control (MMBRC), defined as

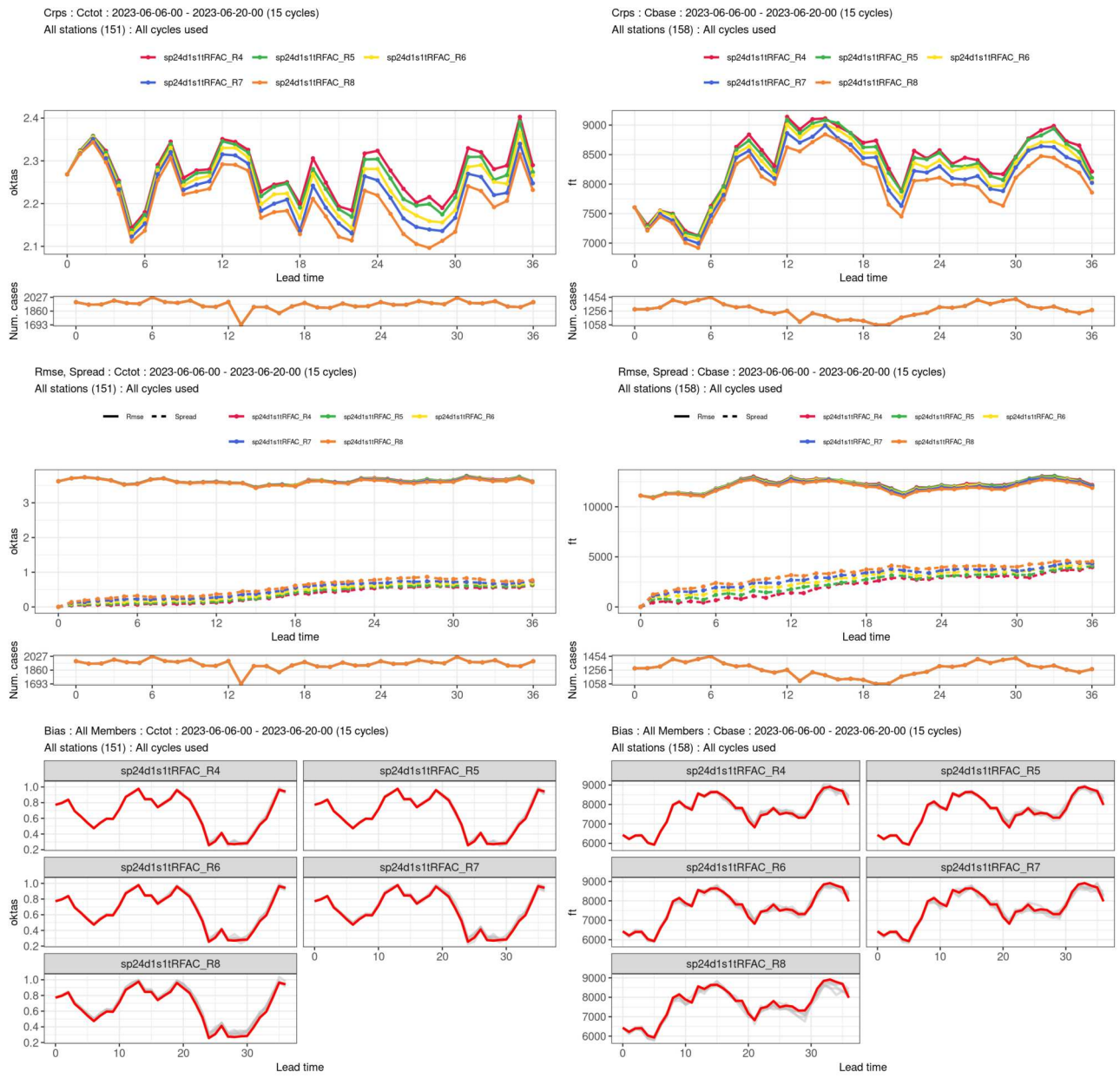


Figure 8: Results from the RFAC_TWOC reference runs for total cloud cover (left) and cloud base (right) over the two-week summer period. CRPS, spread-skill, and member bias are given on the top, middle, and bottom rows. Experiments R4 (ref), R5 (green), R6 (yellow), R7 (blue), and R8 (organge) correspond to $\sigma_R = 0.1$, 0.6, 1.2, 1.8, and 2.4, respectively. See text for further experiment details.

$$\text{MMBRC}\,(l,v) = \frac{1}{N_m}\sum_{j=1}^{N_m}\left[b\,(j,l,v) - b\,(0,l,v)\right] \tag{4}$$

where $b\,(j,l,v)$ represents the bias of ensemble member j at lead time l for verification parameter v (this notation is used henceforth), and N_m is the number of perturbed members i.e. 6 in this case. Note that the relative difference is considered, such that biases for different ensemble members can offset one another. A positive/negative value for MMBRC therefore means that on average the perturbed members are positively/negatively biased relative to the control. The cost functions are based solely on total cloud cover and averaged over the last 24-hours of the forecast, i.e.

$$F = \langle S(l, \text{CCtot}) \rangle = \frac{1}{25} \sum_{l=12}^{36} S(l, \text{CCtot}) \tag{5}$$

where $\langle . \rangle$ denotes the lead time averaging over hours 12-36 and $S(l, v)$ represents a verification metric (at lead time l for parameter v).

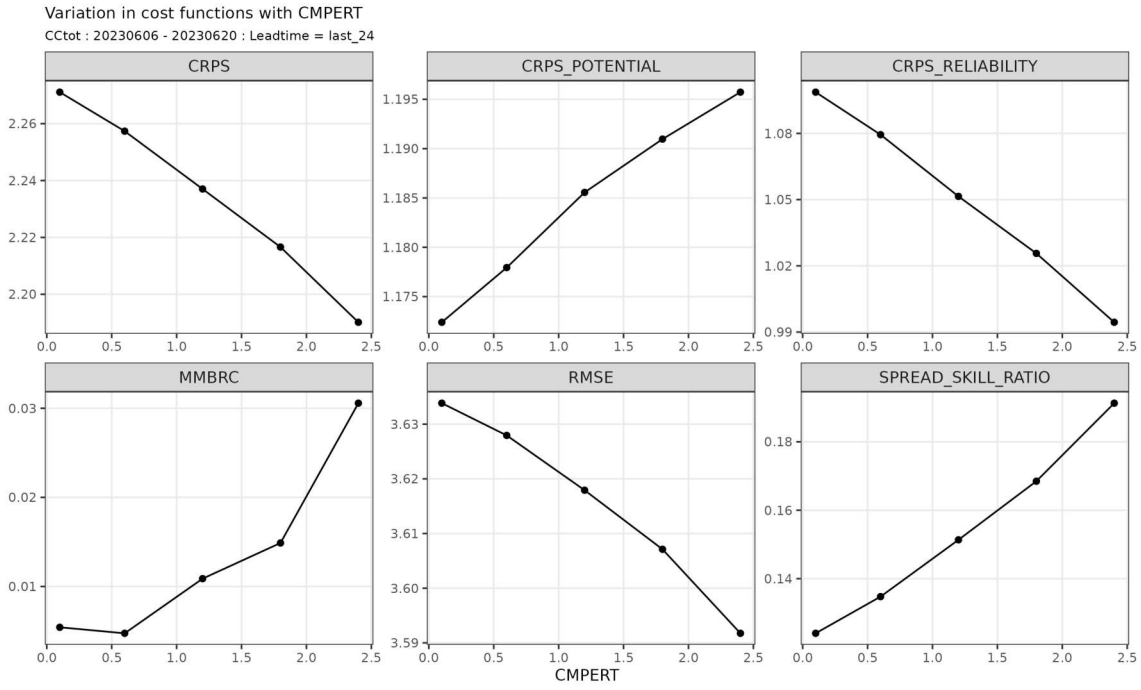


Figure 9: F (defined by (5)) as a function of σ_R for the RFAC_TWOC reference experiments (i.e. R4-R8 in Figure 8). Different choices for the verification score S are indicated by the facet label. Results from the full two-week period are used and scores are averaged over the last 24-hours of the forecast. Note that the units for each cost function are different and hence not indicated on the y-axis.

Figure 9 of course merely reflects what was concluded from Figure 8 but in a more condensed fashion; the "best" values for CRPS, RMSE, and SSR appear to be achieved by taking the maximum value for σ_R over the range considered. The decomposition of CRPS into it's "potential" and "reliability" parts, i.e. $\text{CRPS} = \text{CRPS}_{\text{pot}} + \text{CRPS}_{\text{rel}}$ (Hersbach, 2000), is also given and demonstrates that the improvement in reliability more than compensates for the slight degradation in potential. Finally, the MMBRC demonstrates again that the members become slightly positively biased relative to the control at large σ_R (by ≈ 0.03 oktas on average).

Based on Figure 9, if we were to use one of the traditional "headline" scores for ensemble model performance, say $F = \langle \text{CRPS}(l, \text{CCtot}) \rangle$ or $\langle \text{SSR}(l, \text{CCtot}) \rangle$, as the cost function in the EGO for σ_R, we would anticipate an optimal value of $\sigma_R = 2.4$. This mirrors previous results on this topic, where the "optimal" perturbation standard deviation σ_p based on model SSR or CRPS tends to be the maximum permissible value for σ_p. This is reasonable given that the SPP perturbations can produce reasonable spread without substantially degrading mean RMSE. However optimising based on these cost functions alone would ignore the potentially significant impact SPP can have on perturbed member biases (Tsiringakis et al., 2024). Indeed, much of the manual SPP tuning within HIRLAM and ACCORD has generally sought to find a balance between improving model spread or CRPS without introducing systematic member biases.

There are many ways to construct a cost function which aims to strike this balance. In keeping with the simplistic approach taken to the optimisation, a simple metric is proposed here based on the existing verification metrics generated by harp. First, to assess the added benefit of the SPP perturbation on overall ensemble performance, we define the CRPS RELative to the control Mean Absolute Error (CRPS_REL_MAE) as:

$$\text{CRPS\_REL\_MAE}\,(l,v) = \frac{100 \times (\text{CRPS}\,(l,v) - \text{MAE}\,(0,l,v))}{\text{MAE}\,(0,l,v)} \tag{6}$$

where $\text{MAE}\,(0,l,v)$ represents the control member MAE. The CRPS_REL_MAE is therefore negatively oriented (i.e. the more negative the better). To assess the relative importance of any member bias relative to the control, we can define a scaled version of the MMBRC as

$$\text{MMBRCS}\,(l,v) = \frac{100}{N_m \times \text{MAE}\,(0,l,v)} \sum_{j=1}^{N_m} \left[b\,(j,l,v) - b\,(0,l,v)\right]. \tag{7}$$

This scaling of the member bias relative to the control MAE is useful as it allows one to express if any member bias is of much physical significance e.g. if the control MAE for 2 m temperature was around 1.5 K, and the SPP perturbation introduced a mean member bias of around 0.015 K relative to the control, typically this would not be a major concern. Furthermore, using the control MAE is beneficial compared to the control bias as it naturally avoids any bias ≈ 0 issues. Larger absolute values for MMBRCS are worse.

Finally, we can combine these two metrics together to define

$$\text{CF\_CRPSMB}\,(v) = \langle \text{CRPS\_REL\_MAE}\,(l,v)\rangle + \max\left(\left|\langle \text{MMBRCS}\,(l,v)\rangle\right|^q, \left|\langle \text{MMBRCS}\,(l,v)\rangle\right|\right). \tag{8}$$

Note that:

1. CF_CRPSMB is negatively oriented.
2. The cost function is based on the lead time averages of the separate components, not the lead time average of the sum (although this could equally be done). The former was used here merely to help distinguish the contributions from CRPS and MMBRC.
3. The lead time average of the MMBRCS is used. As such, positive/negative biases at different lead times may compensate one another to give a small value for $\langle \text{MMBRCS}\,(l,v)\rangle$. However this was deemed reasonable as our primary interest for SPP tuning is systematic member biases relative to the control i.e. where members are consistently positively or negatively biased. An example of this is given in Section 3.2.1.
4. The $\max$ is introduced for cases where $|\langle \text{MMBRCS}\,(l,v)\rangle| < 1$ and $q > 1$ (although this is not particularly important given that such small values for the member bias will likely be irrelevant).

The power q is a user-specified tuning parameter; $q = 1$ will balance any percentage benefit in CRPS against any member bias, while $q > 1$ places more weight on avoiding the introduction of a member bias. The behavior of CF_CRPSMB for several values of q is indicated in Figure 10. A value of $q = 3/2$ in (8) was used throughout in this note.

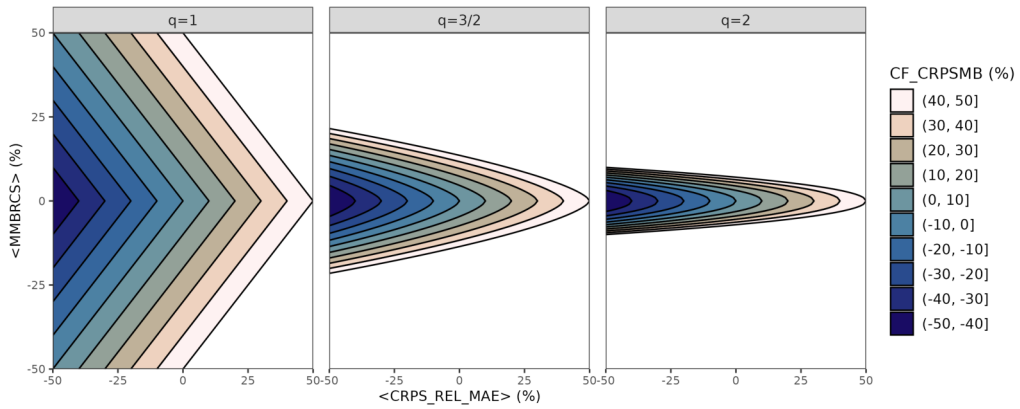


Figure 10: Behaviour of CF_CRPSMB, as in (8), for several values of q (as indicated by the facet label). White indicates values above 50%.

The behaviour of CF_CRPSMB for the RFAC_TWOC reference tests is illustrated in Figure 11 for both total cloud cover and cloud base. Verification statistics for two periods are considered; using the first forecast start date only (i.e. 2023/06/06/00) and using the full two-week period. Using only a single forecast start date in the cost function evaluation is of course not recommended in general, but is used here as part of technical testing. Also indicated is the product of CRPS potential and reliability, i.e.

$$\text{CRPS\_PXR} = \text{CRPS}_{\text{pot}} \times \text{CRPS}_{\text{rel}}, \tag{9}$$

which is another possible choice for the cost function. It may be beneficial to use this CRPS product over CRPS itself as it avoids any direct cancellation of terms in the CRPS and more harshly penalises a degradation in forecast skill (Tuppi et al. (2020) and Pirkka Ollinaho, personal correspondence).

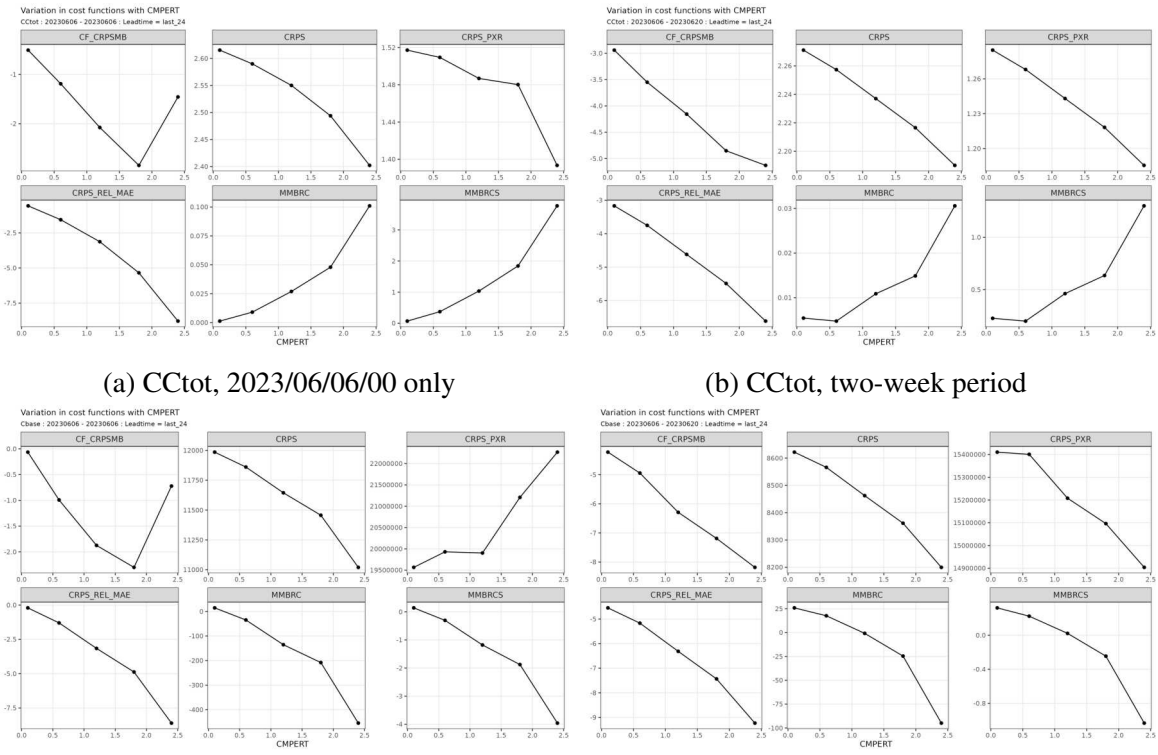
Focusing first on the single cycle results, the CF_CRPSMB cost function has a non-trivial minimum (i.e. not at the extrema) over the range of σ_R considered for both parameters. This is due to the development of a small but not insignificant member bias at large σ_R. However this behaviour is not evident when considering the full two-week period where CF_CPRSMB would again suggest an optimum σ_R of 2.4. In this case there is little to no significant systematic member bias (i.e. MMBRCS is limited to $\approx 1\%$) while the CPRS is improved substantially (i.e. by $\approx 5\%$ relative to the control MAE).

Of course one could carry out much more analysis of possible cost functions using different combinations of verification parameters and statistics, however for our purposes it is sufficient to utilise the simple cost functions in Figures 9 and 11 and assess the performance of the EGO scheme.

3.1.3 Convergence testing

A useful sanity check to assess if the EGO implementation with URANIE in HARMONIE-AROME is working reasonably is to perform convergence tests where the optimum values for the input parameter space (e.g. $X = \sigma_p$) are known a priori. The convergence cost function, F_c, for such tests can take the form

$$F_c = |F - F_{\text{ref}}|, \tag{10}$$



(a) CCtot, 2023/06/06/00 only

(b) CCtot, two-week period

(c) Cbase, 2023/06/06/00 only

(d) Cbase, two-week period

Figure 11: As in Figure 9 but with different cost functions (see facet label). Cost functions based on total cloud cover and cloud base alone are on the top and bottom rows, respectively. The left column is based on verification statistics from a single forecast start date (2023/06/06/00), while the right column uses the full two-week period. CF_CRPSMB, CRPS_REL_MAE, and MMBRCS are percentages, while the other cost functions are in oktas and feet for CCtot and Cbase, respectively.

where F_{ref} represents some cost function for a reference experiment where the input parameter space (i.e. X_{ref}) is specified. As such, if a global minimum for F_c exists (e.g. if F is a monotonic function of X) the EGO scheme should return X_{ref} in order to minimise F_c.

We now test this for the single parameter EGO of σ_R. A single forecast start date (2023/06/06/00) is used to reduce computational costs, the reference data is taken from the experiments outlined in Section 3.1.2, and the possible range for σ_R is $[0.1 - 2.4]$. Initial convergence testing (see examples in Appendix 6.4) demonstrated

1. the importance of using a sensible stopping criteria in the EGO scheme, and
2. a significant benefit to using "setHasMeasurementError(True)" when finding the optimal hyper-parameters for the Kriging.

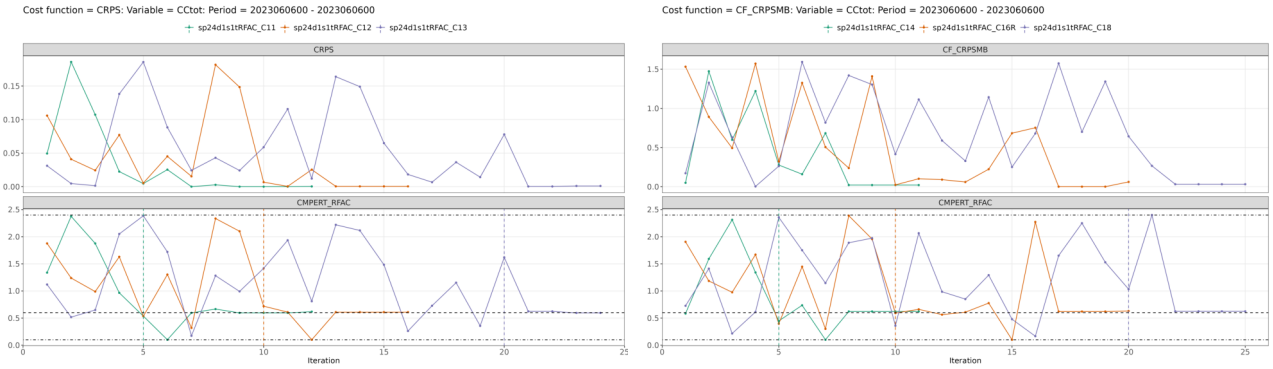
These will be assumed going forward, where we define the EGO stopping criteria as

$$\max(\mathbf{x}) - \min(\mathbf{x}) < (0.05) \times M_x, \quad \mathbf{x} = \{x_k, x_{k-1}, x_{k-2}, x_{k-3}, x_{k-4}\}, \quad \forall x \in X, \tag{11}$$

i.e. for each input parameter x the range of its last five iterations ($\mathbf{x}$) must be less than than 5% of its maximum possible value (M_x). Again this is a very simple choice, with the threshold of 5% chosen

somewhat arbitrarily, but it was found to be effective in terminating the EGO algorithm once it had settled close to "optimised" values for X.

Sample convergence testing results for σ_R are given in Figure 12(a). Here the cost function F used in F_c (10) is the total cloud cover CRPS, i.e. $F = \langle \text{CRPS}\,(l, \text{CCtot})\rangle$, the reference experiment used a value of $\sigma_R = 0.6$, and three initial sample sizes ($N_T = 5, 10, 20$) are considered. In each case the EGO scheme predicts a σ_R value close to the reference quite quickly after completing the initial sample, while the stopping criteria (11) terminates each experiment within $5 - 10$ iterations after N_T. Note that the last predicted value for σ_R has an error of less than 3% (see Table 3). The impact of using a larger initial sample is relatively small in this case, however this is likely due to the simplicity of the optimisaiton problem i.e. one input parameter.



(a) $F = \langle \text{CRPS}\,(l, \text{CCtot})\rangle$ (b) $F = \langle \text{CF\_CRPS}\,(l, \text{CCtot})\rangle$

Figure 12: Variation in the convergence cost function F_c (top row) and predicted value for σ_R (bottom row) as a function of the number of URANIE iterations. Coloured lines indicate different experiments with $N_T = 5$ (green), 10 (orange), and 20 (purple). The cost function used is indicated in the caption. The minimum and maximum possible values for σ_R are indicated by the dot-dash lines, while the reference value for σ_R is given by the horizontal dotted line. The vertical coloured dashed lines indicate N_T for each experiment. See text for further information.

Figure 12(b) illustrates the same set of convergence testing experiments but now using CF_CRPSMB as the cost function i.e. $F = \langle \text{CF\_CRPS}\,(l, \text{CCtot})\rangle$. Quite reasonable behaviour is again observed, with the error in the predicted σ_R limited to 5% (see Table 3) and convergence (as assessed by the stopping condition) within 10 iterations of N_T. Finally, in Table 3 we indicate the results obtained from analogous experiments using a different reference value i.e. $\sigma_R = 1.8$. The error is again limited to $\approx 5\%$.

3.1.4 Optimisation testing

We can now repeat the testing outlined above but in optimisation mode i.e. the convergence cost function (10) is replaced simply by the cost function F. The reference data in Figure 11 can of course be used to provide reasonable estimates for the optimal σ_R for different F, thus providing another sanity check for the EGO performance. For simplicity we again restrict ourselves to total cloud cover and the 2023/06/06/00 cycle only; hence when using CRPS and CF_CRPSMB in $F = \langle S\,(l, \text{CCtot})\rangle$ the optimal σ_R should be approximately 2.4 and 1.8, respectively, based on Figure 11(a).

The results of several optimisation tests are given in Figure 13. They are generally in good agreement with our expectations from the reference runs; σ_R is maximised immediately when using CRPS,

F	N_T	Reference σ_p	Predicted σ_p
CRPS	5	0.6	0.618
CRPS	5	0.6	0.616
CRPS	10	0.6	0.609
CRPS	20	0.6	0.595
CF_CRPSMB	5	0.6	0.621
CF_CRPSMB	10	0.6	0.630
CF_CRPSMB	20	0.6	0.624
CRPS	5	1.8	1.735
CRPS	10	1.8	1.759
CF_CRPSMB	10	1.8	1.713

Table 3: Summary of convergence testing results for σ_R. Each cost function is based on total cloud cover only. The column "Predicted σ_p" represents the last value predicted before the experiment is terminated by the stopping condition. Rows with the same F, N_T, and "Reference σ_p" are repeated experiments.

.

while for CF_CRPSMB it converges to a value of ≈ 1.77 for both $N_T = 5$ and 10. Of course the same optimisation tests could also be run using a longer verification period (e.g. the two week period in Figure 11)(b)), but this was deemed an unnecessary use of resources for such technical testing (see Section 3.2.2 for optimisation over a longer period).

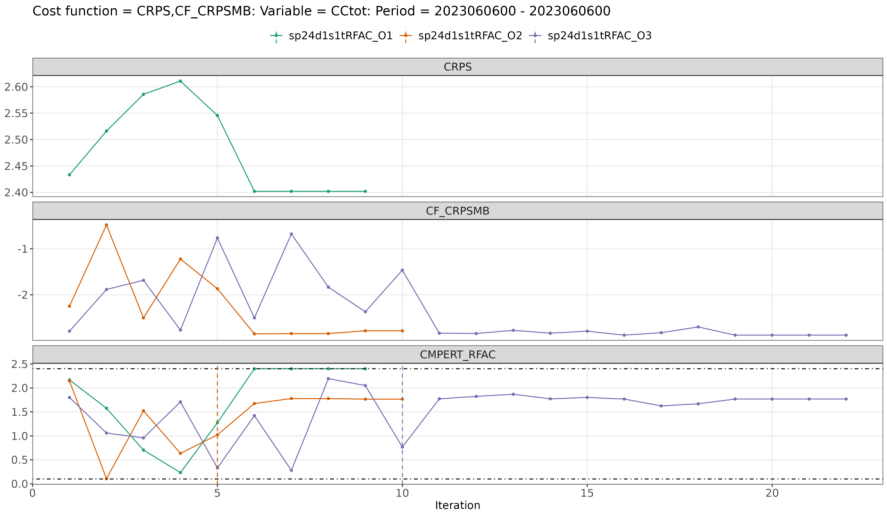


Figure 13: Variation in the cost function F (top and middle rows) and predicted value for σ_R (bottom row) as a function of the number of URANIE iterations. Coloured lines indicate different experiments with CRPS and $N_T = 5$ (green), CF_CRPSMB and $N_T = 5$ (orange), and CF_CRPSMB and $N_T = 10$ (purple). See the text and the caption of Figure 12 for more information.

Overall the the technical convergence and optimisation results for σ_R provide a reasonable level of confidence that the EGO scheme and it's implementation in HARMONIE-AROME is working reasonably (at least for this simple one parameter optimisation). Further tests were also carried out using different cost functions (e.g. using cloud base instead of total cloud cover), but these all corroborated the behaviour described herein. As such, we now move on from technical testing to a more realistic application of the EGO scheme.

3.2 SLWIND

We perform an analogous set of experiments to those described in Section 3.1 but for the SLWIND σ_p, henceforth denoted as σ_S. Investigations within HIRLAM have shown that this SPP parameter can have a very significant impact on ensemble spread, particularly for 10 m wind-speed. As this parameter has only been recently introduced into HARMONIE-AROME, it's optimal configuration is still to be determined. Hence it constitutes an obvious candidate for EGO testing.

The experiment configuration follows that described in Section 3.1.1 but with SLWIND instead of RFAC_TWOC active, i.e. the SPP settings are as described in Table 2 with

- $d_p = 0$ (default value for $V(M)$),
- $\sigma_p = \sigma_S$ is the single variable to be optimised using EGO,
- a uniform distribution with $o_p = 0.5$, and
- perturbation clipping set to $[-0.4, 0.4]$.

Possible SLWIND perturbation distributions for various σ_S are illustrated in Figure 14. Here we consider quite a large range for σ_S (i.e. 0.01 - 0.8) in order to fully explore the impact of this perturbation at more extreme values, and this range will be used henceforth unless otherwise stated. Note that the clipping limits of $[-0.4, 0.4]$ effectively means that no clipping is applied for $\sigma_S \leq 0.8$. This was chosen in order to maintain a uniform distribution for all σ_S considered.

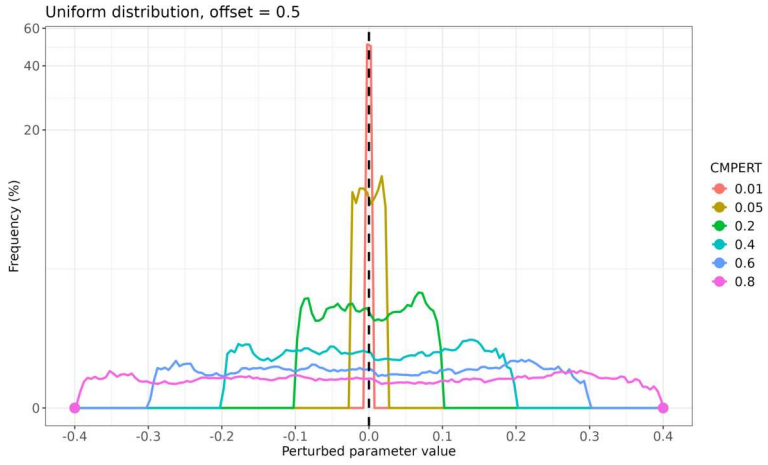


Figure 14: SPP perturbation distributions for SLWIND using values indicated in Table 2 with $d_p = 0$, $o_p = 0.5$, and clipping of $[-0.4, 0.4]$. The colors correspond to different values for σ_S, while the R_p used is a sample taken from a HARMONIE-AROME experiment.

Finally, in addition to the two-week summer period discussed previously, a 10-day winter period (2022/02/10/00 - 2022/02/20/00, with 5-day spin-up) was also considered for the reference tests. This period included a number of named storms over Ireland, and hence was used to assess the performance of SLWIND for high winds.

3.2.1 Reference runs

Sample point verification results demonstrating the impact of σ_S over the summer and winter periods are given in Figures 15, 16, and 18 for PMSL, 10 m wind speed, and 2 m temperature, respectively. For each parameter once can observe that SLWIND has a very significant impact on ensemble spread without significantly degrading ensemble mean RMSE for the two verification periods. Commensurately there is a clear and consistent improvement in CRPS as σ_S increases in all cases.

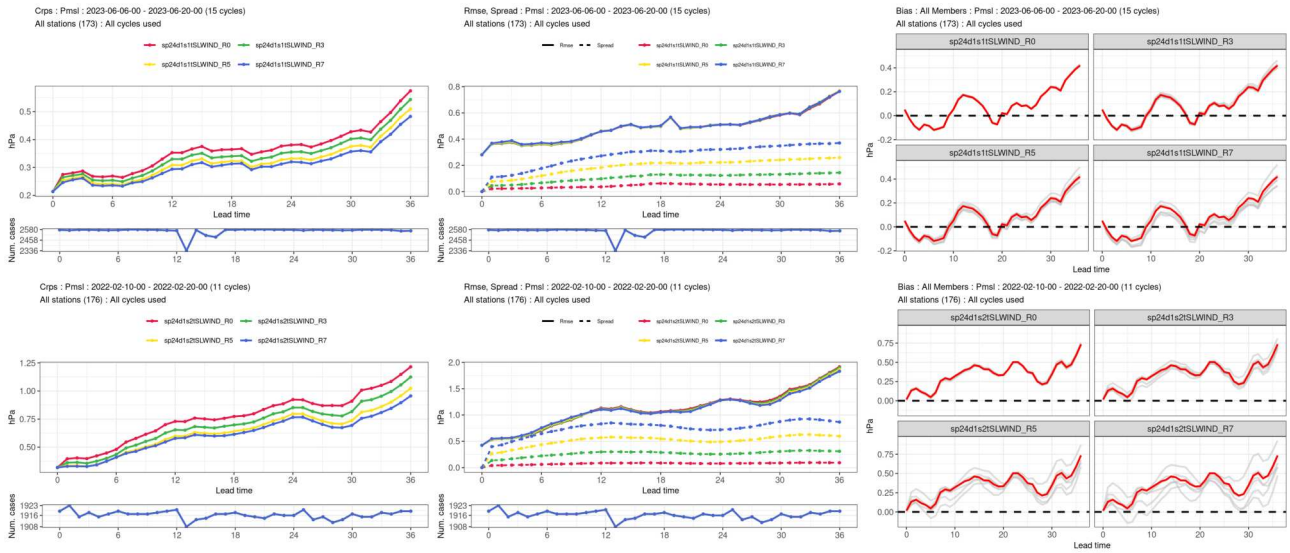


Figure 15: PMSL results from the SLWIND reference runs for summer (top row) and winter (bottom row) periods. CRPS, spread-skill, and member bias are given on the left, middle, and right columns. Experiments R0 (red), R3 (green), R5 (yellow), and R7 (blue) correspond to $\sigma_S = 0.05$, 0.2, 0.4, and 0.6 respectively. See text for further experiment details. Note the use of different scales on the y-axes.

Focusing on the member biases relative to the control, there is a signal for a negative PMSL bias at larger values of σ_S, which is consistent across both seasons (right column of Figure 15). For 2 m temperature, there is a consistent cooling of the members relative to the control during winter, adding to the existing model cold bias. A signal for cooling at large σ_S during the summer period is also evident, however this is offset by a slight warm bias during hours 20-30 (top-right of Figure 18).

For 10 m wind speed however, the member bias behaviour differs between the two seasons. For the summer period, the larger σ_S introduces a consistent positive wind speed bias in the pertubed members relative to the control, while for winter the control is much more centered. It's possible that this is related to the performance of the model for the contrasting meteorological conditions during the two periods. The summer period consisted of quite light winds (< 5 m/s on average) with a clear diurnal cycle which the model consistently underestimates (see Figure 17), while for winter the model generally overestimated the higher winds associated with storms on February 16$^{\text{th}}$, 18$^{\text{th}}$, and 20$^{\text{th}}$.

These verification results are condensed into selected cost functions as a function of σ_S in Figure 19, where lead time averaging over the last 24-hours of the forecast is again used. Cost functions for five model parameters are illustrated along with the mean of the cost function over all parameters, i.e.

$$F_{5p} = \frac{1}{5} \sum_{v \in V} \langle S(l, v) \rangle , \quad V = \{\text{Pmsl}, \text{S10m}, \text{T2m}, \text{Td2m}, \text{CCtot}\} . \tag{12}$$

For each model parameter and period we find that CRPS_REL_MAE and the SSR are monotonic

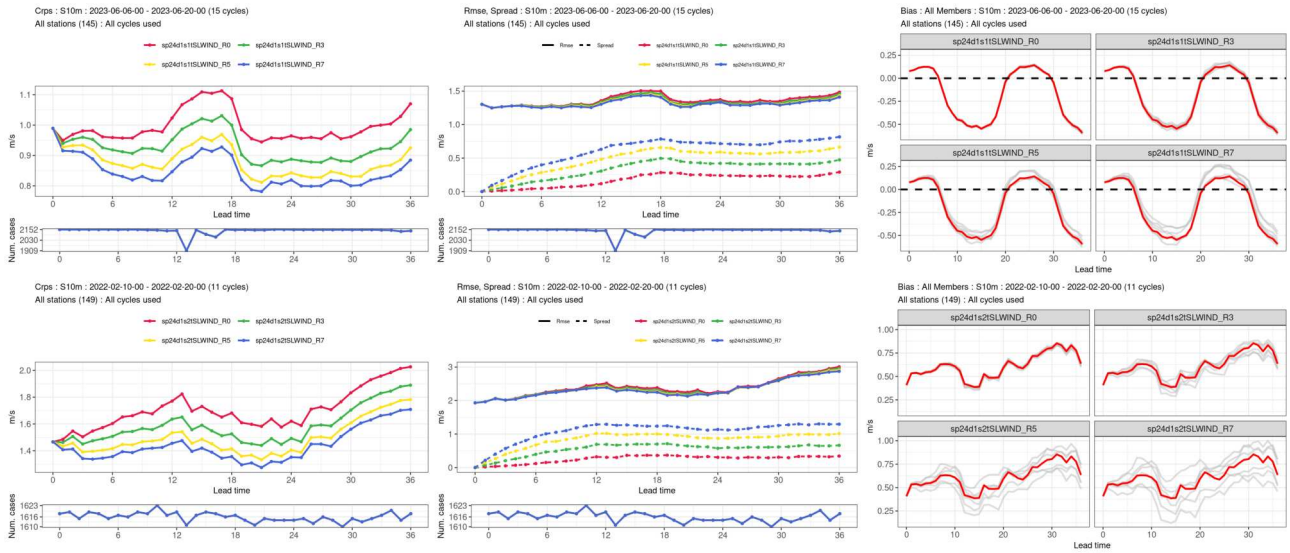
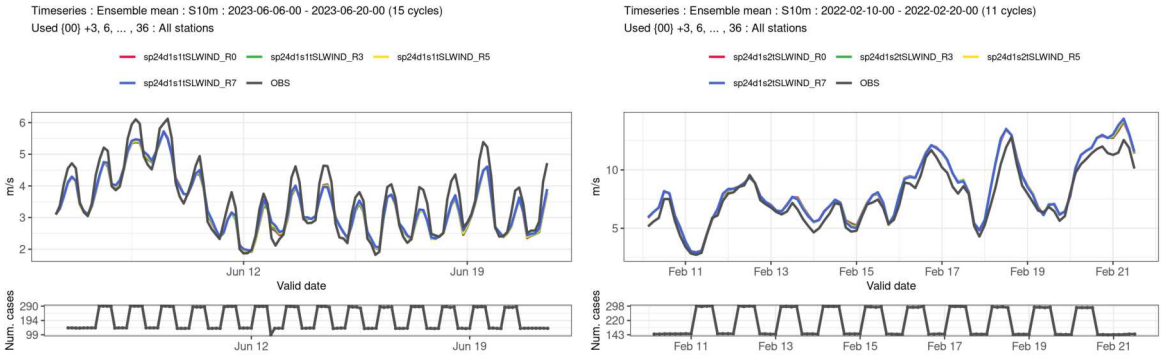


Figure 16: As in Figure 15 but for 10 m wind speed.



(a) Summer (b) Winter

Figure 17: Ensemble mean 10 m wind speed over the summer and winter verification periods. Observations are given in black. See the caption of Figure 15 for a description of the experiment names. Note the use of different scales on the y-axes.

increasing/decreasing functions of σ_S over the range considered. These scores also illustrate that SLWIND has the largest positive impact on total cloud cover, particularly so at relatively modest values of σ_S (e.g. 0.2). The MMBRCS reflects the member bias behavior previously discussed; a significant positive wind speed bias is observed for the summer period but not in winter, while a negative PMSL bias at large σ_S is consistent over the two periods. Note that for the summer period the 2 m temperature MMBRCS is close to zero for all σ_S considered, and this reflects the impact of lead time averaging over offsetting perturbed member biases (top-right of Figure 18).

As such for most parameters the CF_CPRSMB has a non-trivial minimum over the σ_S range considered, and this is also reflected in the mean over all five parameters as defined in (12). While using such a parameter mean is a very coarse way to combine the parameter cost functions, it demonstrates very similar behavior over the two periods tested with an optimum value of $\sigma_S \approx 0.4$.

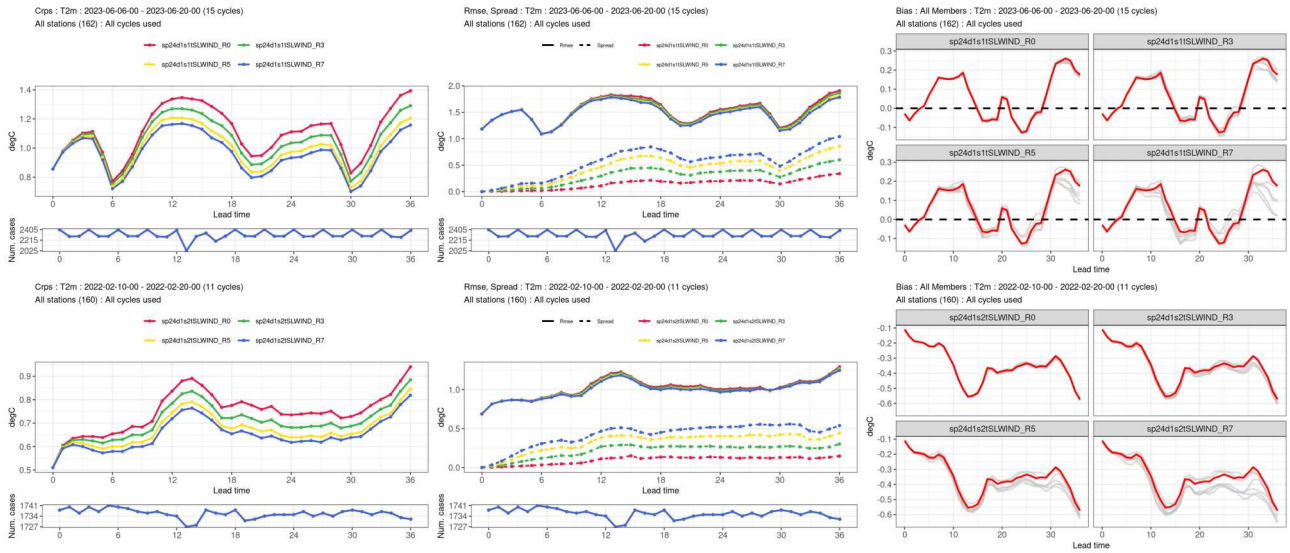
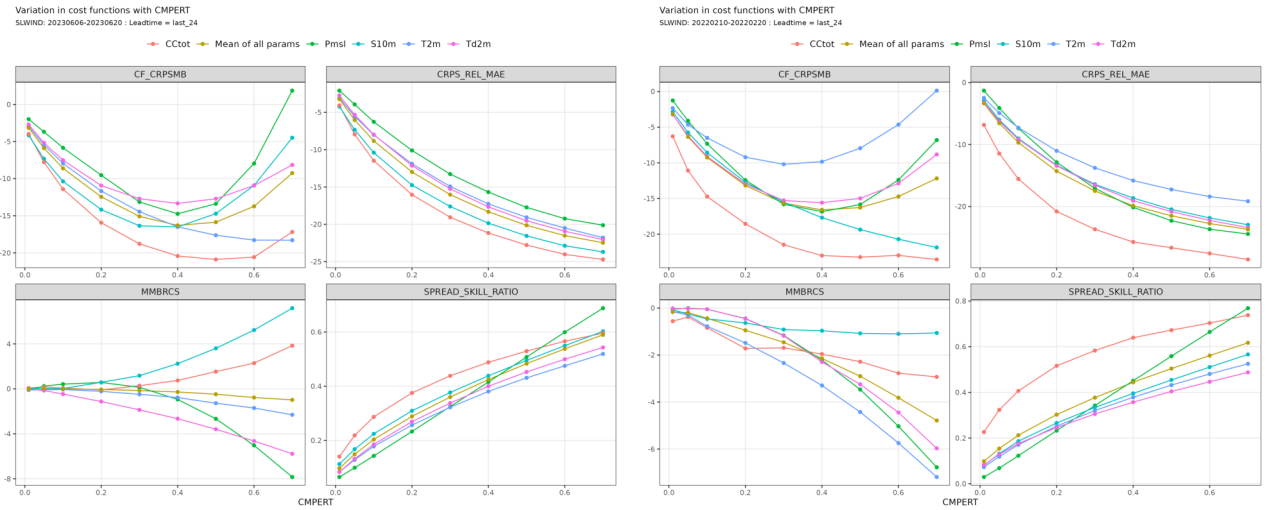


Figure 18: As in Figure 15 but for 2 m temperature.



(a) Summer

(b) Winter

Figure 19: Various cost functions as a function of σ_S for the SLWIND reference experiments over the two verification periods (using all forecast start dates). Each line represents $\langle S(l,v)\rangle$ for that parameter except for "Mean of all params" which is given by (12).

3.2.2 Convergence and optimisation results

The reference experiments are now used as a guide for several convergence and optimisation experiments with the EGO scheme for σ_S. A number of single cycle convergence tests (again using the 2023/06/06/00 cycle and $F = \langle \text{CF\_CRPSMB}\,(l, \text{Pmsl})\rangle$ as the cost function) were carried out to assess functionality. Results are summarised in Table 4, which generally corroborate the convergence results presented in Section 3.1.3 and give further confidence in the scheme's performance.

The convergence tests using a reference value of $\sigma_S = 0.2$, however, clearly struggle to recover the correct value. This is due to the fact that the convergence cost function has two global minimum points due to the shape of the cost function (see Appendix 6.5), and the scheme can end up oscillating between these minima (see Figure 20). This reflects one of the shortcomings of these simple conver-

gence tests. Several single cycle optimisation tests for this experiment configuration were found to quickly converge to an optimum value of $\sigma_S \approx 0.1$ (not shown), which is again in good agreement with what one would expect based on the reference experiments (see Appendix 6.5).

N_T	Reference σ_p	Predicted σ_p
10	0.05	0.055
5	0.1	0.08
10	0.1	0.098
10	0.2	0.045
10	0.2	0.035
10	0.3	0.301
10	0.3	0.293

Table 4: Summary of single cycle convergence tests for σ_S for the 2023/06/06/00 cycle with $F = \langle \text{CF\_CRPSMB}\,(l, \text{Pmsl})\rangle$. Note that σ_S was restricted to the range $[0.01, 0.4]$ in these tests. See the caption of Table 3 for more information.

.

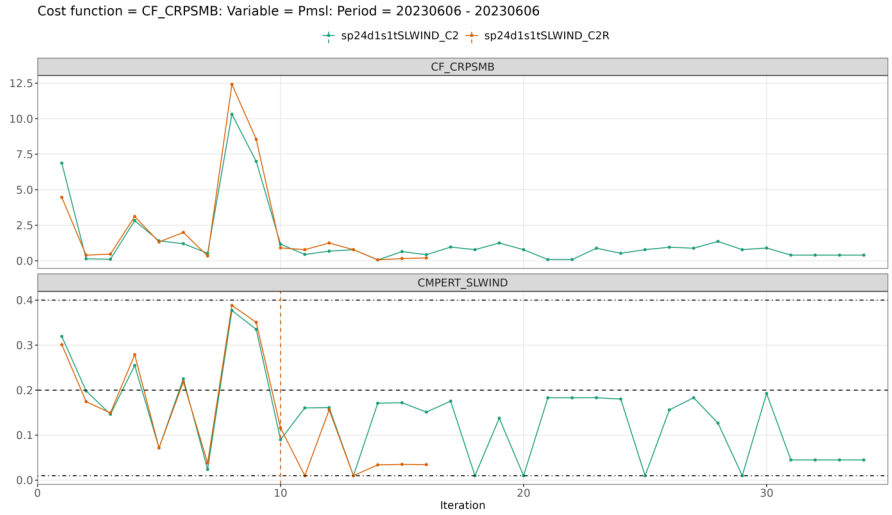
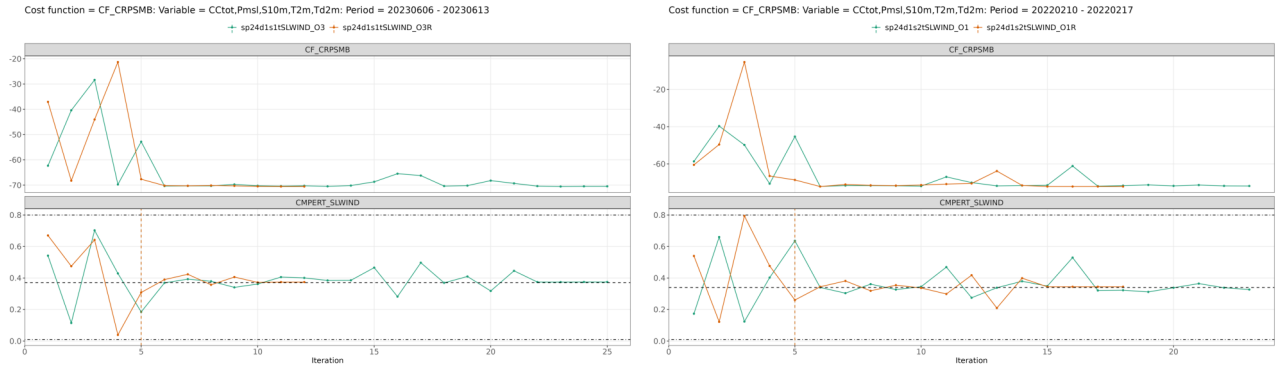


Figure 20: Convergence cost function (top row) and predicted σ_S (bottom) iteration series. Coloured lines indicate two experiments with identical configurations apart from the initial sample values. The cost function used is $F = \langle \text{CF\_CRPS}\,(l, \text{Pmsl})\rangle$ for the 2023/06/06/00 cycle only. σ_S was restricted to the range $[0.01, 0.4]$ with a reference value of 0.2. See the caption of 12 for more information.

As a more realistic assessment of the EGO scheme for σ_S tuning, optimisation runs over week-long verification periods in summer and winter were carried out. A single week (i.e. the first seven of each period) was used as opposed to the full 14- and 10-day periods, respectively, to limit computational costs (as discussed in Section 2.4). This restriction to a single week was found to have relatively little impact on the qualitative behaviour of the cost function dependence on σ_S as presented in Figure 19 (see Appendix 6.5). With this limitation, the long optimisation runs presented here typically cost on the order of 3M SBU.

In an effort to arrive at a value of σ_S which gives the "best" ensemble performance for a range of parameters, the five parameter mean cost function F_{5p}, defined by (12), with $S = \text{CF\_CRPSMB}$ was used in these long optimisation tests. An initial sample size of $N_T = 5$ was used based on the



(a) Summer (b) Winter

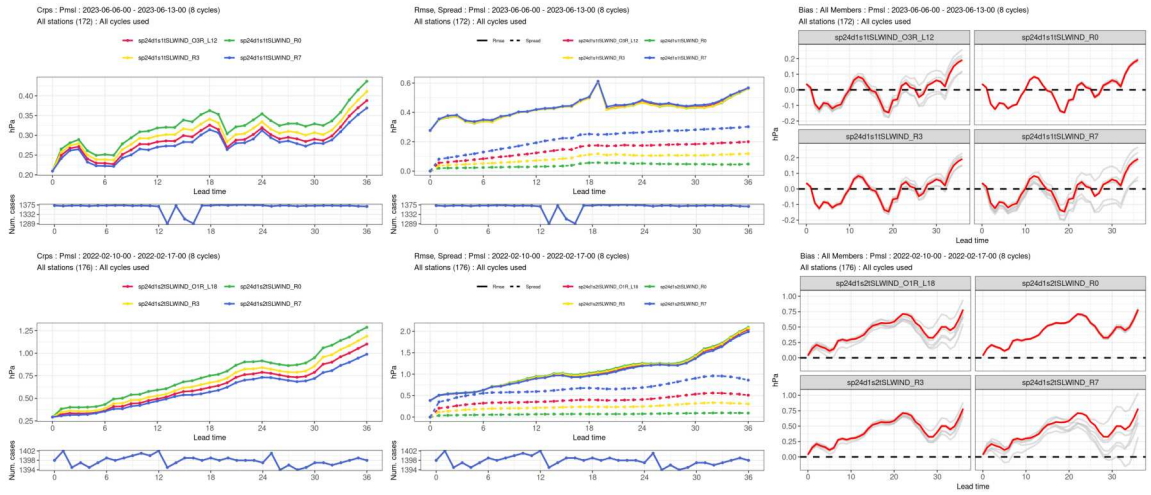
Figure 21: Variation in F_{5p} with $S =$ CF_CRPSMB (top row, see (12)) and the predicted value for σ_S (bottom row) for the week-long optimisation runs. Coloured lines indicate different experiments with identical configurations apart from the initial sample values. The dashed horizontal lines correspond to 0.37 and 0.34 for summer and winter, respectively. See the text and the caption of Figure 12 for more information.

reasonable performance of the EGO single cycle tests for one parameter optimisation. The outcome of these tests are given in Figure 21, where for each period the same experiment is run twice in order to assess the reproducibility of the predicted optimum σ_S.

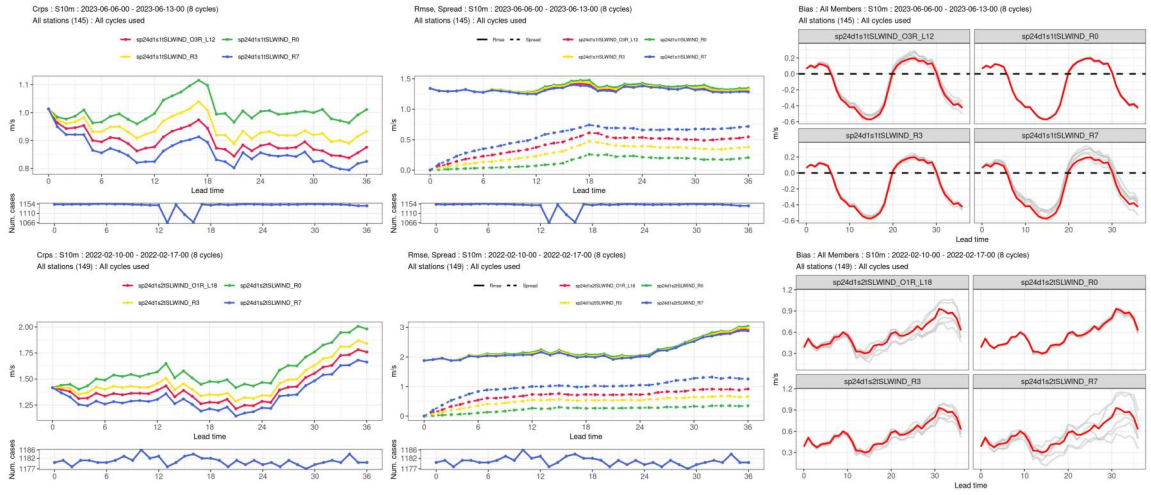
Overall the long optimisation runs perform quite reasonably. Convergence to a reproducible optimum value is obtained within 25 iterations for both periods considered. In this particular case the predicted optimum σ_S is very similar in summer an winter, namely 0.37 and 0.34 respectively, however one would not necessarily expect the same optimum across different verification periods in general. These predicted optimal values are in good agreement with the cost function analysis of the reference runs in Section 3.2.1 (see also Appendix 6.5).

Finally, in Figure 22 we compare the verification results obtained using these predicted optimal σ_S versus the reference runs discussed previously. Note that the optimal σ_S data for each period comes directly from the last URANIE iteration of the corresponding optimisation run (i.e. the last iteration of each red line in Figure 21). When compared to the default value for σ_S, i.e. 0.05, most verification statistics for each parameter look quite favourable for the predicted optimal σ_S. One area of possible concern however is the member bias; while the 10 m wind biases look reasonable, the pertubed members are slightly negatively biased relative to the control for Pmsl and 2 m temperature during the winter period. Of course this could be addressed by modifying the cost function used in the optimisation, e.g. using a larger value for q in (8).

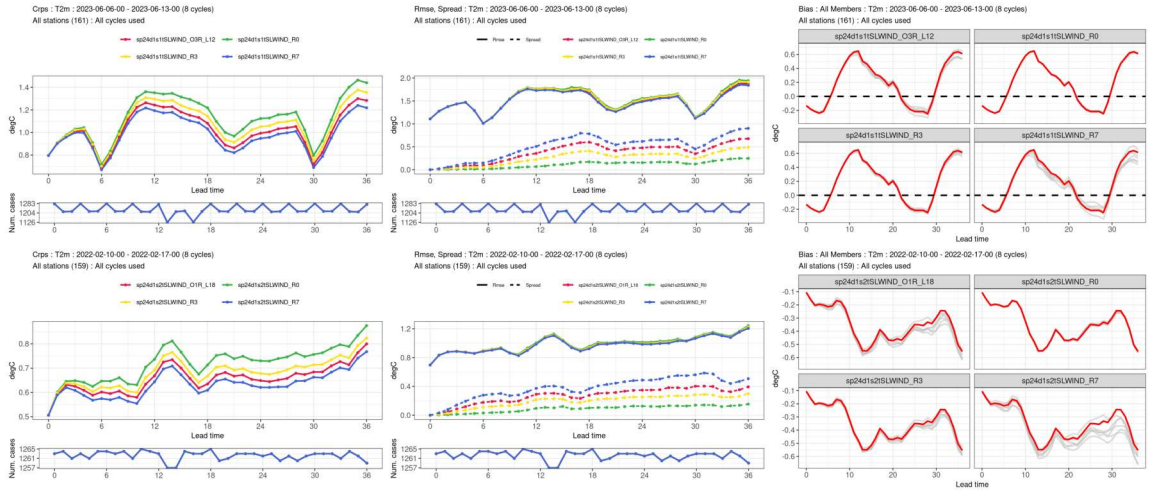
Overall the results presented in this section appear relatively promising. The functionality of the EGO workflow in HARMONIE-AROME has been confirmed through analysis of the reference and convergence runs, a possible cost function which balances CRPS improvement against member biases has been introduced, and the "realistic" long optimisation runs perform quite reasonably. While the experiments presented here are of course very simple, the results suggest that this approach could be suitable for automatic single parameter SPP tuning in the future.



(a) Pmsl



(b) 10 m wind



(c) 2 m temperature

Figure 22: Comparison of verification results using the "optimal" σ_S, as determined in Figure 21, and the reference runs. The top and bottom row in each sub-figure correspond to a week-long summer and winter period, respectively. CRPS, spread-skill, and member bias are given on the left, middle, and right columns. The red line uses the optimal σ_S (≈ 0.37 and ≈ 0.34 for summer and winter, respectively) while green, yellow, and blue use $\sigma_p = 0.05$, 0.2, and 0.6, respectively.

4 Two parameter optimisation

Based on the relatively promising results for the EGO scheme in HARMONIE-AROME for optimisation of a single SPP configuration parameter, we now briefly assess it's performance for a slightly more complicated two parameter optimisation problem. Optimisation of multiple SPP parameters at the same time would significantly benefit the task of SPP tuning within HIRLAM/ACCORD and represent a significant step forward. The focus here will remain on optimising σ_p but now for two SPP parameters simultaneously; namely RFAC_TWOC and SLWIND. These parameters are a natural choice in order to leverage on previous experience.

The experiment configuration and parameter settings are as previously outlined in Sections 3.1.1 and 3.2 unless otherwise stated, except that both the RFAC_TWOC and SLWIND SPP perturbations are active at the same time.

4.1 Reference runs

We again perform a series of reference runs to act as benchmarks for further convergence and optimisation testing. Effectively sampling the two-dimensional space $[\sigma_S, \sigma_R]$ is of course costly and time consuming. As such we restrict interest to a single forecast start date (i.e. 2023/06/06/00) and take $\sigma_S \in [0.01, 0.05, 0.1, 0.2, 0.3, 0.4]$ and $\sigma_R \in [0.1, 0.6, 1.2, 1.8, 2.4]$, thus yielding 30 reference experiments to run. Note the reduced upper-bound used for σ_S compared to Section 3.2. In terms of the cost function analysis, our main focus will again be CF_CRPSMB.

We first consider the behaviour of $\langle \text{CF\_CRPSMB}\,(l, \text{Pmsl}) \rangle$ based on these two-dimensional reference runs in Figure 23. Here each dot represents an individual experiment with $[\sigma_S, \sigma_R]$ as indicated. Results for the individual RFAC_TWOC and SLWIND reference experiments presented in Section 3 (i.e where only one SPP parameter is active) are also included on the x/y axes. For this particular experiment configuration, the scatterplot would suggest that the cost function is optimised at $[\sigma_S, \sigma_R] \approx [0.2, 2.4]$ for the range of $[\sigma_S, \sigma_R]$ considered.

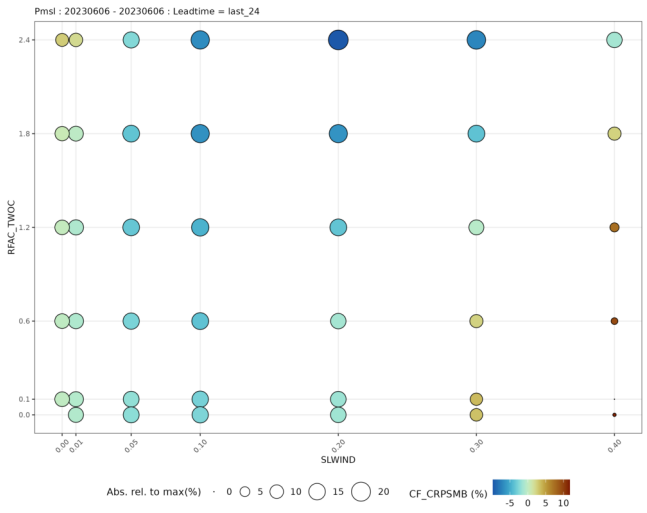
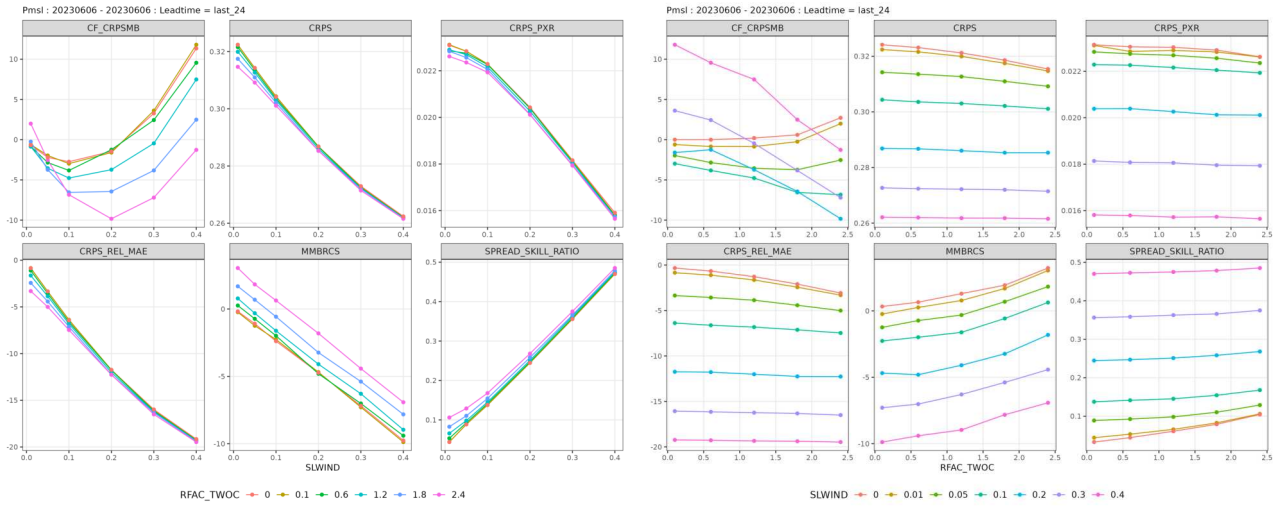


Figure 23: Variation of $F = \langle \text{CF\_CRPSMB}\,(l, \text{Pmsl}) \rangle$ in $[\sigma_S, \sigma_R]$ parameter space based on the reference experiments. Dot colour indicates the value of F (i.e. blue is better) and dot size is $|F - \max(F)|$ (i.e. larger the better). Results for $\sigma_S = 0$ and $\sigma_R = 0$ are taken the from single parameter reference runs in Section 3. Data based on a single forecast start date of 2023/06/06/00.

It is interesting to note the clear interaction between the two SPP parameters in this particular case, For example, the introduction of the RFAC_TWOC perturbation alters the optimal value for σ_S when compared to running SLWIND on its own, i.e. the optimal σ_S increases from 0.1 for $\sigma_R = 0$ to 0.2 for $\sigma_R = 2.4$. In order to investigate this in more detail, we plot the cost function as a function of σ_R and σ_S separately in Figure 24. The impact of RFAC_TWOC on the Pmsl CPRS and SSR is found to be quite small overall in this case, with a marginal improvement in these scores as σ_R increases (as expected). However this parameter does introduce a positive perturbed member bias for Pmsl as it increases, and this positive bias acts to somewhat offset the significant negative bias introduced by the SLWIND perturbation (i.e. a case of compensating errors). As such the lower values for the cost function at larger σ_R reflect a reduction in systematic member bias as opposed to a direct improvement in CRPS.



(a) Versus σ_S

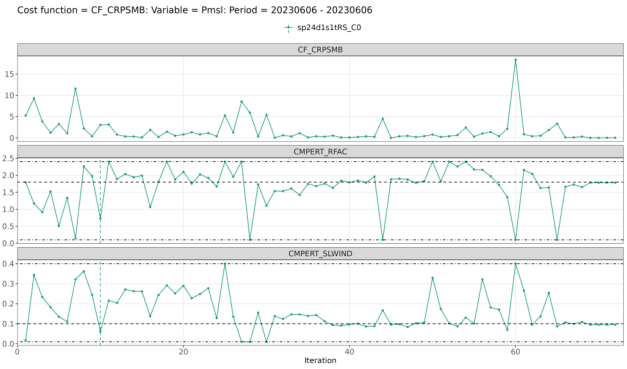
(b) Versus σ_R

Figure 24: As in Figure 23 but as a function of σ_S and σ_R individually. Other cost functions are also included, as indicated by the facet label. The coloured lines indicates different fixed values for σ_S or σ_R.

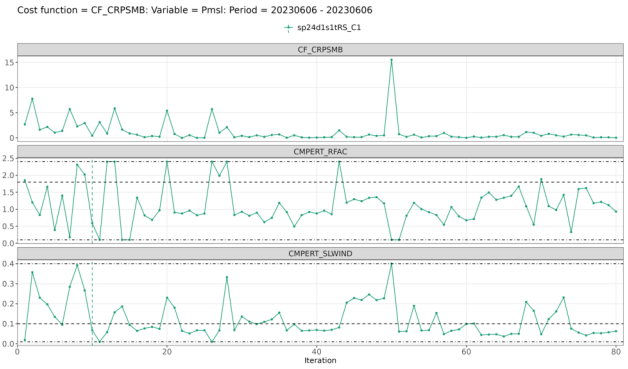
The same analysis for total cloud cover is given in Appendix 6.6. In this case, SLWIND has a considerable impact on CRPS without introducing a member bias, while RFAC_TWOC introduces a positive member bias at large σ_R. The combination of these effects thus leads to optimal values of $[\sigma_S, \sigma_R] \approx [0.4, 0.1]$ for the range considered. If one considers the five parameter mean F_{5p} defined by (12), the optimal values are found to be $[\sigma_S, \sigma_R] \approx [0.3, 2.4]$ (not shown).

4.2 Convergence and optimisation results

Indicative results from several two-parameter convergence experiments are given in Figure 25, using the cost function $\langle \text{CF\_CRPSMB}\ (l, \text{Pmsl}) \rangle$ for the 2023/06/06/00 cycle. Two sets of reference $[\sigma_S, \sigma_R]$ are considered and the initial sample size used was 10. In one case we find that the scheme eventually recovers the reference values after approximately 70 iterations, while in the other case the scheme drifts without convergence for both input parameters over 80 iterations ($N_{\max}$ in this case). This is not particularly surprising given the increased likelihood for multiple global minima points as the number of input parameters increases. As such, these simple convergence tests are of less practical use when considering multi-parameter optimisation problems.



(a) $[0.1, 1.8]$ reference

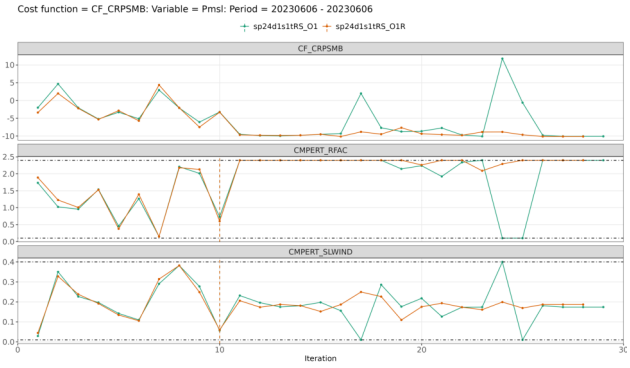


(b) $[0.2, 1.2]$ reference

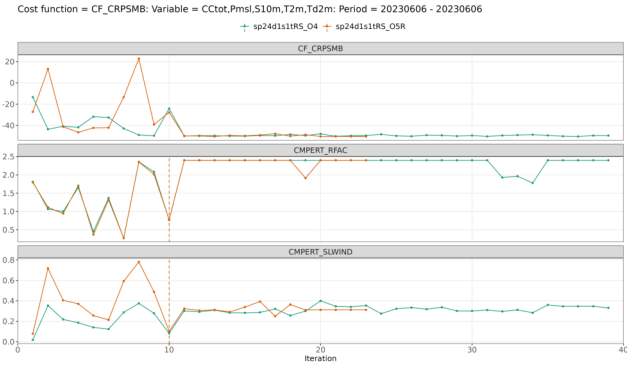
Figure 25: Convergence cost function (top row) and predicted σ_R (middle) and σ_S (below) iteration series. The cost function used is $F = \langle \text{CF\_CRPS}\,(l, \text{Pmsl})\rangle$ for the 2023/06/06/00 cycle only. σ_S and σ_R restricted to the range $[0.01, 0.4]$ and $[0.1, 2.4]$ with reference values indicated in the caption.

In Figure 26 we assess the performance of the EGO scheme for several single-cycle two-parameter optimisation experiments with $N_T = 10$. Focusing first on $\langle \text{CF\_CRPS}\,(l, \text{Pmsl})\rangle$, the scheme appears to perform reasonably well with convergence within 30 iterations. The predicted optimal values for $[\sigma_S, \sigma_R]$ are also found to be relatively reproducible, with $[0.174, 2.4]$ and $[0.187, 2.4]$ predicted by two experiments which are identical apart from the initial sample values used. These results are in good agreement with the qualitative analysis from the reference experiments in Figure 23. This was also found to be the case when using $\langle \text{CF\_CRPS}\,(l, \text{CCtot})\rangle$, with a predicted optimum of $[0.4, 0.1]$ (not shown).

Similar performance can be observed when using F_{5p} as the cost function (Figure 26(b)). In this case the two experiments considered are identical apart from the upper bound for σ_S, with values of 0.4 and 0.8 considered. The predicted optimal values are again very similar in both cases; namely $[0.331, 2.4]$ and $[0.313.2.4]$. One can note a significant difference in the number of iterations until the experiment terminates i.e. 39 vs 23 for an upper bound of 0.4 and 0.8 respectively, despite σ_S being relatively constant in both experiments. This likely reflects the fact that the stopping condition (11) scales with the maximum possible value for the input parameter.



(a) $\langle \text{CF\_CRPS}\,(l, \text{Pmsl})\rangle$



(b) F_{5p} with $S = \text{CF\_CRPSMB}$

Figure 26: Cost function (top row) and predicted σ_R (middle) and σ_S (below) iteration-series for single-cycle optimisation runs. In (a), coloured lines indicate different experiments with identical configurations apart from the initial sample values. In (b), the experiments are identical apart from the upper bound for σ_S, where green and orange uses 0.4 and 0.8, respectively.

Moving on to a more realistic test, in Figure 27 we consider the results from a week-long optimisation experiment for $[\sigma_S, \sigma_R]$ over the summer period. Here an upper bound of 0.8 is used for σ_S (for consistency with the single parameter optimisation in Figure 20) and $N_T = 10$. The observed behaviour is somewhat disappointing but not particularly surprising; while the scheme quickly converges to a value of $\sigma_S \approx 0.4$, and generally stays close to this value over the entire simulation, the value of σ_R does not converge and continues to drift between its minimum and maximum possible values. This signals a very weak dependency in the cost function to σ_R, and as such the simulation continues to iterate with little variation in the cost function (i.e. F_{5p} with $S =$ CF_CRPSMB in this case). A notable exception to this is the spike in the cost function at iterations 26-28 where the scheme samples the extrema for $[\sigma_S, \sigma_R]$, a feature which is frequently observed when the cost function has a weak dependency on one of the input parameters.

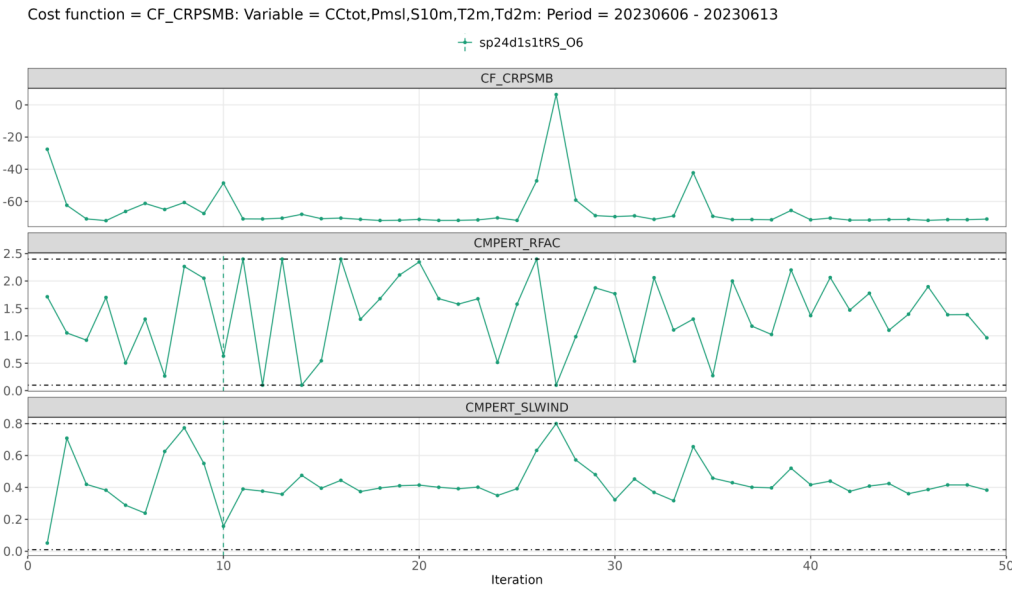
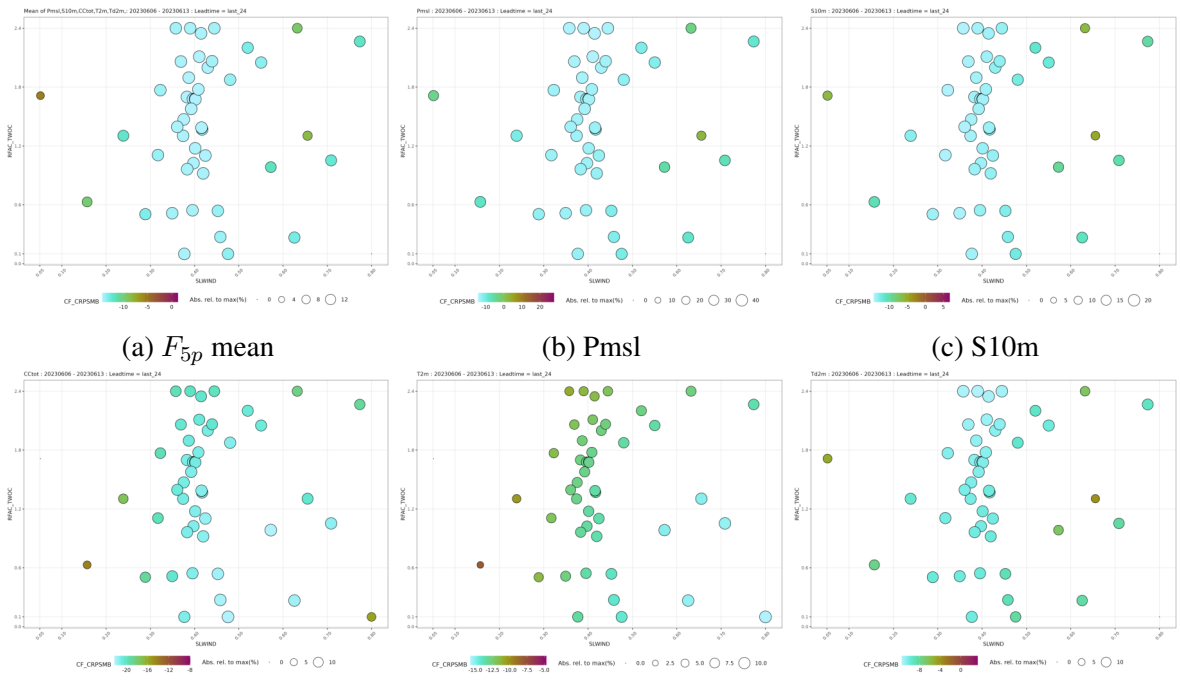


Figure 27: Variation in F_{5p} with $S =$ CF_CRPSMB for the summer week-long optimisation test of $[\sigma_S, \sigma_R]$, with $5 \times F_{5p}$, σ_R, and σ_S on the top, middle, and bottom rows, respectively. See the text for further details.

In Figure 28 each iteration of the experiment is plotted in $[\sigma_S, \sigma_R]$ space, which highlights the clustering of values at $\sigma_S \approx 0.4$ and the weak variation in the five parameter mean F_{5p} over the range of σ_R considered. The CF_CRPSMB contribution from each parameter is also given, which helps to explain why the experiment fails to converge to an optimal σ_R. For Pmsl and S10m the introduction of systematic biases at large σ_S (negative and positive respectively, as observed previously) tends to limit σ_S, with σ_R having little to no impact on either parameter. T2m and Td2m tend to offset one another somewhat, with smaller σ_R tending to improve/degrade the cost function for T2m and Td2m, respectively, while the variation of CCtot in $[\sigma_S, \sigma_R]$ space appears to be non-linear but relatively weak, particularly at $\sigma_S \approx 0.4$. As such, when averaged over the five parameters there is no clear signal for improved performance for any σ_R.

It is also interesting to contrast the model performance for this two parameter experiment, where both SLWIND and RFAC_TWOC perturbations are active, against an experiment where only SLWIND is active. Given that the two parameter optimisation run does not predict a clear optimal value for σ_R, for comparison purposes it is reasonable to use the last set of predicted values, i.e. $[\sigma_S, \sigma_R] \approx [0.38, 0.96]$, as a notional optimum[8]. Sample point verification results are given in Figure 29 which compare this

[8]Valid since the last value of σ_S is close to the value predicted in the majority of the experiment.



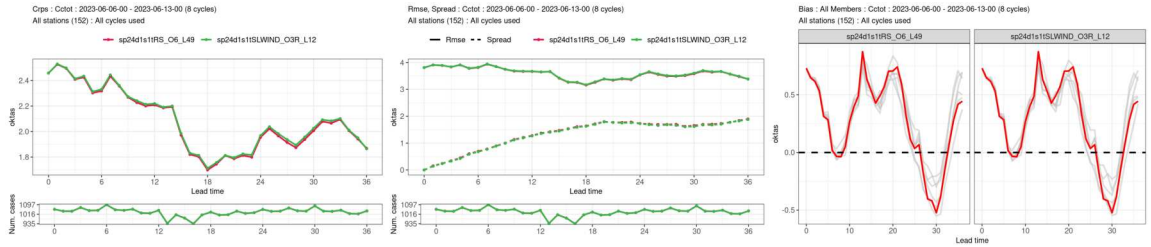
(a) F_{5p} mean (b) Pmsl (c) S10m

(d) CCtot (e) T2m (f) Td2m

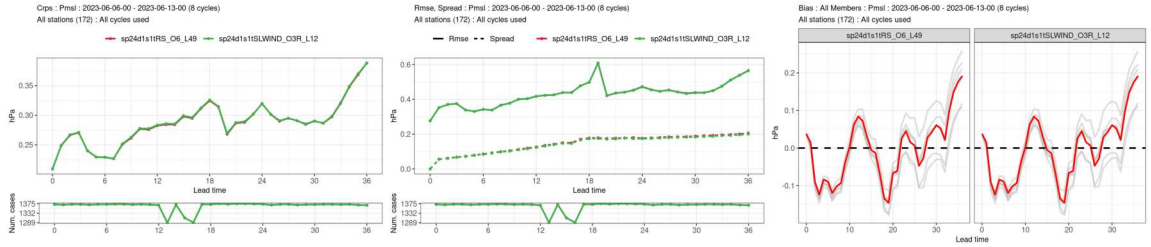
Figure 28: Variation in (a) F_{5p} with $S =$ CF_CRPSMB and (b-f) $\langle$CF_CRPSMB $(l, v)\rangle$, with parameter v indicated in the caption, for the summer week-long optimisation test of $[\sigma_S, \sigma_R]$. See alsi the caption of Figure 27. The colourbar limits are variable in each sub-figure to improve readability.

"optimal" two parameter experiment versus the optimal single parameter SLWIND experiment (as determined in Figure 21). Clearly there is very little impact of adding the RFAC_TWOC perturbation on top of SLWIND, despite the fact that when used on it's own RFAC_TWOC can have an appreciable impact (see Figure 8). This is a commonly observed feature when testing SPP perturbations in combination. One can also note that the predicted value of σ_S from the two parameter optimisation (≈ 0.38), is in good agreement with that observed from the single parameter optimisation (≈ 0.37), which again provides more evidence that the EGO scheme is working reasonably.

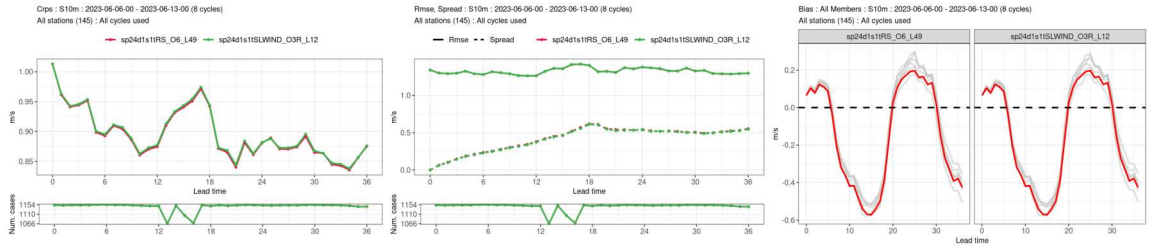
This experiment of course reflects some of the limitations of the approach taken in this note. If one was to optimise a single parameter instead of F_{5p}, e.g. T2m, it's likely that the scheme would converge, and similarly so if a different cost function was used, e.g. the SSR. The experiment also highlights an important missing component from the stopping criteria (11); namely the variation in the cost function itself. In retrospect this is an obvious omission, and in future work the stopping criteria should be modified to check if the cost function is approximately constant over the previous X iterations. Introducing such a check will automatically terminate an experiment such as this where the cost function has a weak dependency on one or more of the input parameters, thus saving computational resources. For example, in Figure 27 the max and min of F_{5p} over iterations 17-25 is -14.04% and -14.37%, respectively. In this case, the experiment was manually terminated at iteration 49 with a cost of $\approx$ 7 M SBUs.



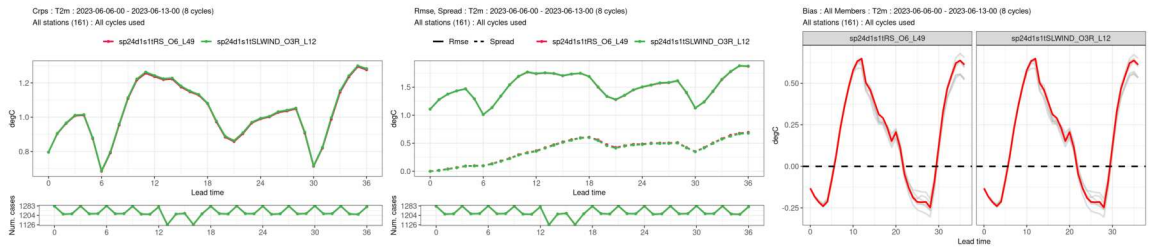
(a) Total cloud cover



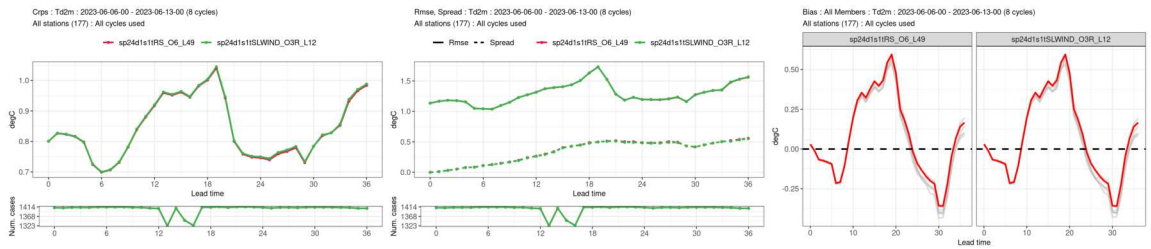
(b) Pmsl



(c) 10 m wind



(d) 2 m temperature



(e) 2 m dew point temperature

Figure 29: Comparison of verification results for the two SPP parameter experiment using the last predicted values for $[\sigma_S, \sigma_R] \approx [0.38, 0.96]$ in Figure 27 (red line), versus the single SPP parameter experiment using the "optimal" $\sigma_S \approx 0.37$ as determined in Figure 21 (green line). CRPS, spread-skill, and member bias are given on the left, middle, and right columns.

5 Conclusions and next steps

This NWP note has taken a very simple approach to the optimisation of one or two HARMONIE-AROME SPP configuration parameters, namely the perturbation standard deviation σ_p, using the URANIE platform and the EGO scheme. Reference tests have been carried out to investigate the impact of σ_p on model performance for different SPP parameters. This impact was quantified using verification statistics, and these statistics were used to propose a simple cost function which balances an improvement in model CPRS against the introduction of a systematic perturbed member bias relative to the control (denoted here as CF_CRPSMB). The main results from the suite of experiments presented herein are summarised below.

1. The technical implementation of the EGO scheme in HARMONIE-AROME with URANIE appears to work correctly (for both one and two parameter optimisation).
2. Raw verification statistics such as spread-skill or CRPS are not particularly well-suited to optimisation of σ_p.
3. Single parameter optimisation of σ_p using CF_CRPSMB can yield quite sensible results. For SLWIND, using the predicted optimal σ_S gives a clear improvement in overall model performance when compared to the default value for σ_S, albeit with some systematic member bias remaining. The optimal σ_p appears to be reproducible and the typical cost of an optimisation run is reasonable (approximately 3 M SBU).
4. While the two parameter optimisation tests also provided reasonable results, the long run highlighted some of the limitations associated with the approach taken when the cost function used has a very weak dependency on some input parameter(s).

As such, with some adaptions the general approach outlined in this note could be applied to wider SPP configuration tuning within HIRLAM/ACCORD relatively soon, at least for single parameter optimisation. Some immediate next steps for future work include:

1. Further experimentation with the cost function used in the EGO algorithm e.g. should something other than CF_CRPSMB be used, should the exponent q be increased, etc.?
2. Correct the stopping criteria to also include a check on cost function variation (see Section 4.2).
3. In retrospect, it would have been more appropriate to use the Fair CRPS, using a reference ensemnble size of infinity, instead of CRPS in the definition of CF_CRPSMB in (8). This is because the Fair CRPS in harp is reproduced exactly with the same input data, whereas this is not the case for CRPS (e.g. running the same verification twice can give CRPS values which differ on the order of 10^{-6}). This is just a small technical correction and should have little to no impact on overall performance.
4. Further long optimisation tests with multiple σ_p for different SPP parameters.
5. Assess simultaneous tuning of σ_p with the offset o_p for uniform distributions, which may help avoid the introduction of member biases.

References

Fannon, J. and Clancy, C. (2024). Utilising the URANIE platform for sensitivity analysis and optimisation of ensemble perturbation methods in the HARMONIE-AROME model. ECMWF Special Project request, https://www.ecmwf.int/sites/default/files/special_projects/2024/spiefann-2024-request.pdf.

Frogner, I.-L., Andrae, U., Ollinaho, P., Hally, A., Hämäläinen, K., Kauhanen, J., Ivarsson, K.-I., and Yazgi, D. (2022). Model uncertainty representation in a convection-permitting ensemble — SPP and SPPT in harmonEPS. *Monthly Weather Review*, 150(4):775 – 795.

Hersbach, H. (2000). Decomposition of the continuous ranked probability score for ensemble prediction systems. *Weather and Forecasting*, 15(5):559 – 570.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492.

Lang, S. T. K., Lock, S.-J., Leutbecher, M., Bechtold, P., and Forbes, R. M. (2021). Revision of the stochastically perturbed parametrisations model uncertainty scheme in the integrated forecasting system. *Quarterly Journal of the Royal Meteorological Society*, 147(735):1364–1381.

Tsiringakis, A., Frogner, I.-L., de Rooy, W., Andrae, U., Alan, H., Contreras Osorio, S., van der Veen, S., and Barkmeijer, J. (2024). An update to the stochastically perturbed parameterization scheme of harmonEPS. *Monthly Weather Review*, 152(8):1923 – 1943.

Tsyrulnikov, M. and Gayfulin, D. (2017). A limited-area spatio-temporal stochastic pattern generator for simulation of uncertainties in ensemble applications. *Meteorologische Zeitschrift*, 26(5):549–566.

Tuppi, L., Ollinaho, P., Ekblom, M., Shemyakin, V., and Järvinen, H. (2020). Necessary conditions for algorithmic tuning of weather prediction models using openifs as an example. *Geoscientific Model Development*, 13(11):5799–5812.

Van Ginderachter, M. (2021). Full-system sized ensemble forecasts within the URANIE framework. ESCAPE-2 project, https://www.hpc-escape2.eu/sites/default/files/2021-11/ESCAPE-2-D4-6-V1-0.pdf.

Van Ginderachter, M. (2022). HarmonEPS VVUQ using the URANIE platform. ACCORD All Staff Workshop 2022, https://www.umr-cnrm.fr/accord/IMG/pdf/uranie-harmoneps_mvg_asw2022.pdf.

Van Ginderachter, M. and Fannon, J. (2024). URANIE: a toolbox for VVUQ. ACCORD EPS Working Week, Budapest, https://opensource.umr-cnrm.fr/attachments/download/5543/20240123_EPS_michielvg.pdf.

6 Appendix

6.1 Experiment input data

Table 5 indicates the location on the ATOS HPC of the input data and binaries used for all experiments in this NWP note. IFSHRES lateral boundary conditions were retrieved from the UWC-W archive on ECFS.

Description	Path
Binaries (gnu)	/ec/res4/hpcperm/dujf/spiefann2024/bin_archive/d46h1eps
Climate files	/ec/res4/hpcperm/dujf/spiefann2024/clm/IRELAND25S
LBCs	/ec/res4/scratch/dujf/uwc_lbc/DINI_HRES
Observations	/ec/res4/scratch/dujf/spiefann2024/obs_IRELAND25S

Table 5: Location of HARMONIE-AROME input data and binaries used.

6.2 EGO settings

Table 6 indicates various settings relevant to the implementation of the EGO algorithm with URANIE.

Description	Value	Relevant URANIE function
Distribution type for initial samples	Uniform	TUniformDistribution
Algorithm for generating initial samples	lhs	TSampling
Correlation function used for Kriging	matern3/2	TGPBuilder
Optimisation criterion for Kriging	LOO	findOptimalParameters
Size of screening DOE for Kriging	500	findOptimalParameters
Optimisation algorithm for Kriging	Subplexe	findOptimalParameters
Max. number of optimisation runs for Kriging	10,000,000	findOptimalParameters
Optimisation algorithm for EI	TNloptCobyla	TNlopt

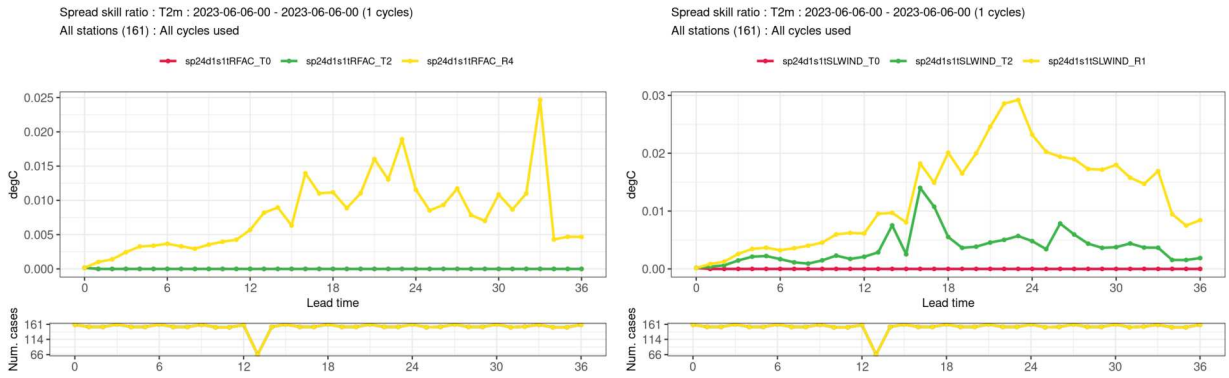
Table 6: EGO-related settings used throughout unless otherwise stated. DOE stands for "design-of-experiments".

6.3 No perturbation tests

Two technical tests were carried out to ensure that the HARMONIE-AROME configuration detailed in Table 1 worked as expected when no model perturbation was applied. In particular:

- SPP perturbations were switched off entirely by setting "SPP=no" in ecf/config_exp.h (this will be referred to as test "T0").
- SPP perturbations were retained (i.e. "SPP=yes") for a single SPP parameter but the standard deviation was set to zero (i.e. $\sigma_p = 0$). According to (1) and (2), this should have the same effect of switching off SPP (this will be referred to as test "T2").

In both cases the perturbed members should collapse to the control member and the ensemble spread should be zero. Sample point verification results for the spread-skill ratio are illustrated in Figure 30. When the SPP parameter considered is RFAC_TWOC, the spread reduces to zero as expected for tests T0 and T2. One can note, however, that non-zero spread is observed with $\sigma_p = 0$ for SLWIND. While negligible, this may point to a small bug in this dynamics perturbation.



(a) RFAC_TWOC (b) SLWIND

Figure 30: Sample 2 m temperature spread-skill ratio results for a single forecast start date and tests T0 (ref) and T2 (green). A reference experiment (yellow), using a uniform distribution with (a) $\sigma_p = 0.1$ and (b) $\sigma_p = 0.01$, is also included for comparison.

6.4 Initial EGO convergence testing

The results of two initial RFAC_TWOC EGO convergence tests, one using CPRS and the other CF_CRPS as the cost function F, are given in Figure 31. Here F is calculated solely from the total cloud cover verification statistics (i.e. $F = \langle S\left(l, \text{CCtot}\right)\rangle$) for the 2023/06/06/00 cycle (see Figure 11). The reference experiment, to which comparison is made in the convergence cost function (10), used a value of $\sigma_p = 0.6$. A large initial sample size of $N_T = 40$ is used and the maximum number of iterations $N_{\max} = 80$.

As illustrated in Figure 31, for the immediate iterations after the initial sampling the EGO algorithm predicts values of σ_p quite close to the reference value of 0.6, as expected. This is the case both for CRPS and CF_CPRS as cost functions. However as the scheme iterates further, the predicted optimum value can deviate significantly, particularly when using CF_CRPSMB as the cost function. This can be reconciled by considering the top left panel of Figure 11(a), where the value of CF_CPRSMB at $\sigma_p \approx 2.4$ is also quite close to CF_CRPSMB at the reference value. Ultimately there is no clear convergence to the reference value in this case even after 80 iterations.

In Figure 32 we repeat the CPRS EGO convergence test described above but now using the "setHasMeasurementError(True)" option when finding the optimal hyper-parameters in the Kriging. Here the same initial training data is used in both experiments to isolate the impact. Turning on this option clearly benefits the convergence behaviour, and indeed the algorithm continues to return a value of ≈ 0.62 upon further iterations. In this case the error in the predicted optimum value is $\approx 3\%$ relative to the reference, which is quite reasonable. Figure 32 also emphasises the importance of including a convergence check to terminate the experiment; otherwise significant computational resources could be wasted iterating up to $N_{\max}$ with very little variation in the predicted value for σ_p.

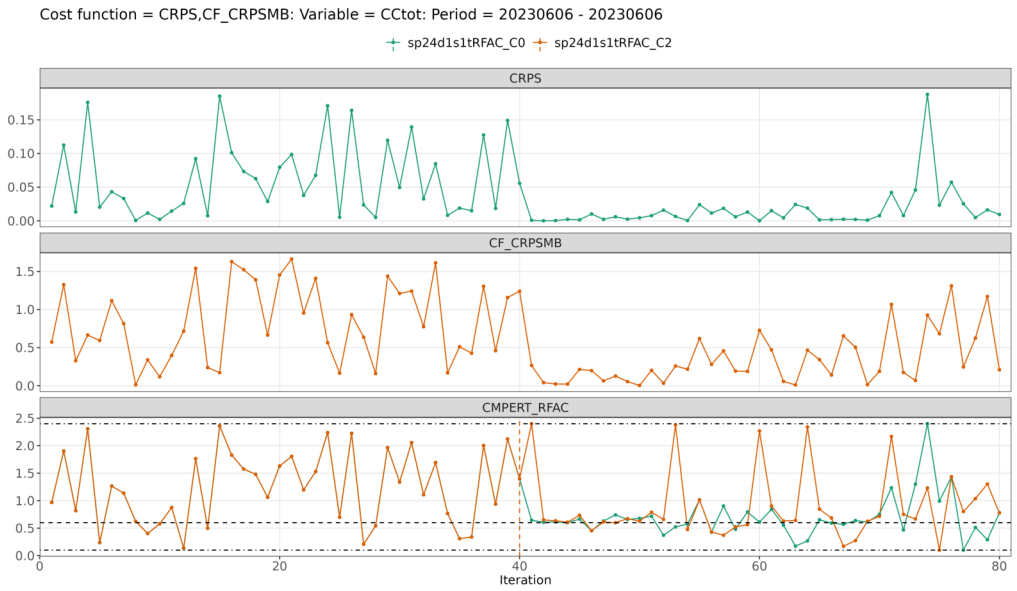


Figure 31: Variation in the convergence cost function F_c (top and middle rows) and predicted value for σ_p (bottom row) as a function of the number of URANIE iterations. Coloured lines indicate different experiments, one using CRPS (green) and the other CF_CRPS (orange) as the cost function F in F_c. The minimum and maximum possible values for σ_p are indicated by the dot-dash lines, while the reference value for σ_p is given by the horizontal dotted line. The vertical coloured dashed lines indicate N_T for each experiment.

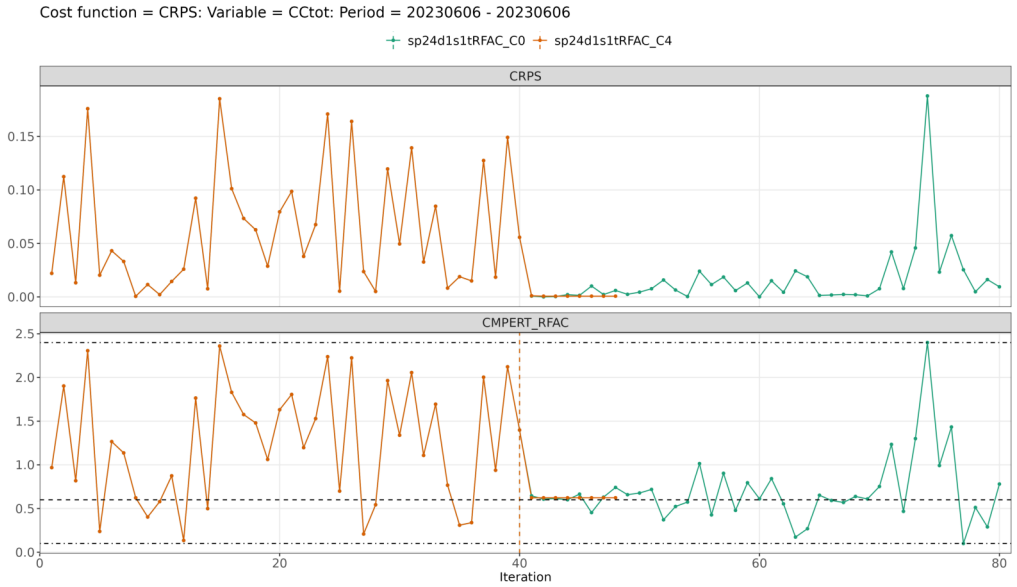


Figure 32: As in Figure 32 but using CRPS only as the cost function. The two experiments are identical apart from the activation of "setHasMeasurementError(True)" (orange line).

Utilising the "setHasMeasurementError(True)" option is appropriate here given that components of the cost functions considered (e.g. mean member bias relative to the control) are estimated using a relatively small number of ensemble members.

6.5 Additional results for Section 3.2

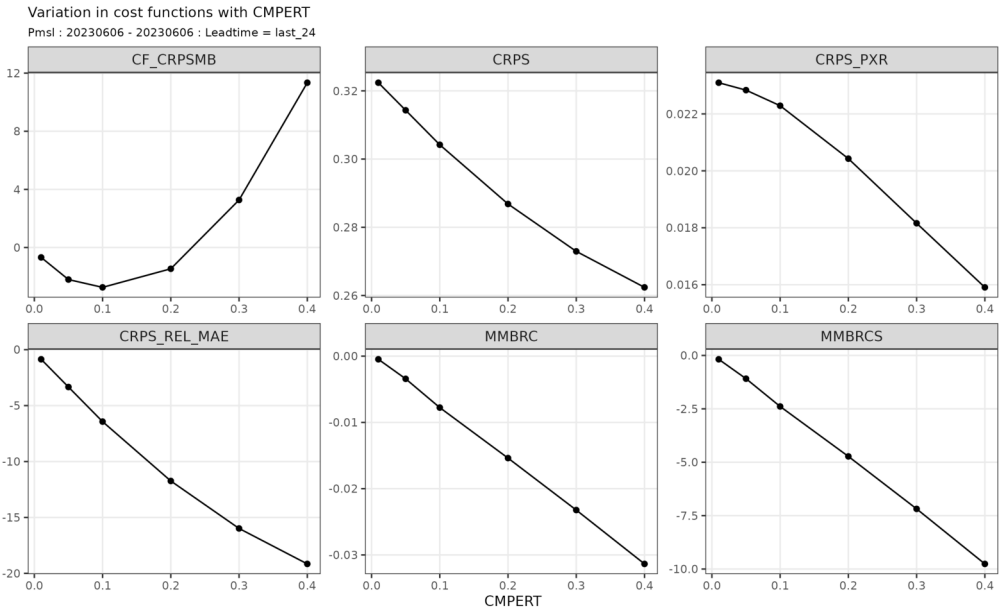
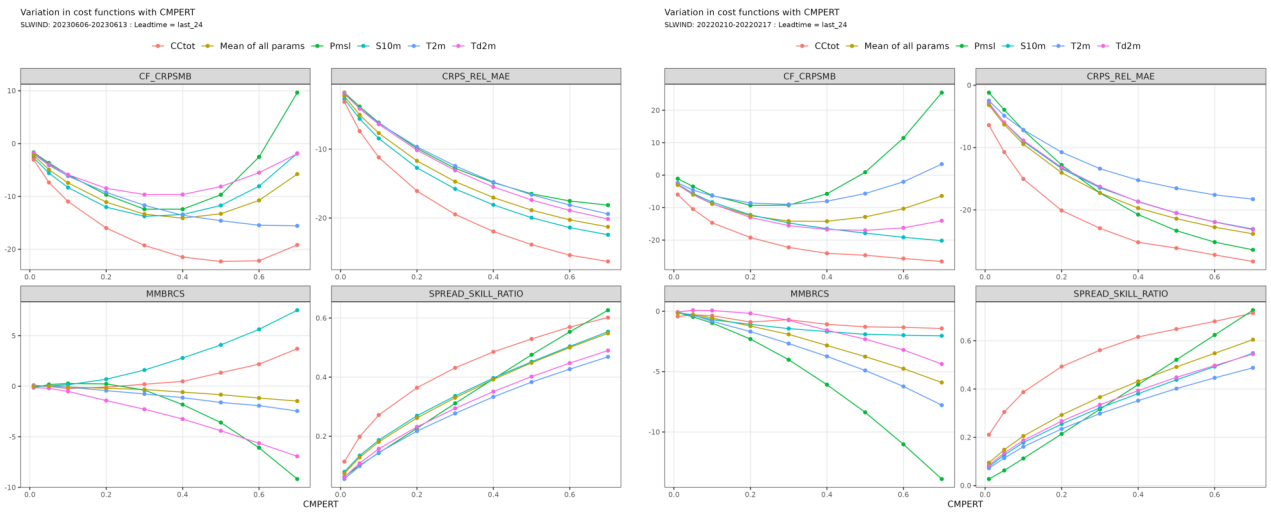


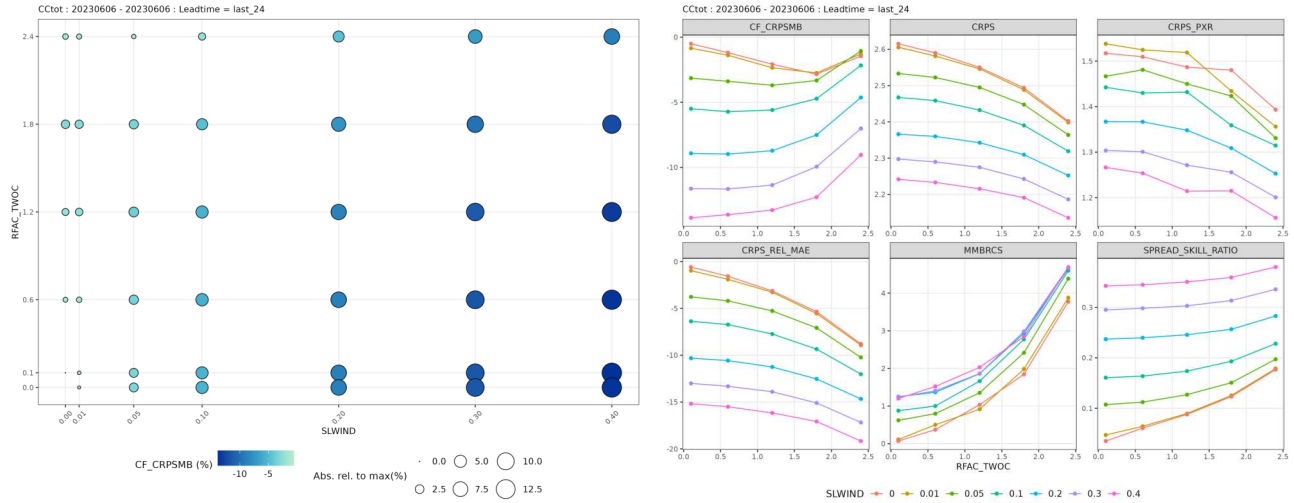
Figure 33: As in Figure 11 but for SLWIND σ_p reference tests and Pmsl.



(a) Summer (b) Winter

Figure 34: As in Figure 19 but only using data from the first seven days of each period.

6.6 Additional results for Section 4



(a) $\left[\sigma_p^S, \sigma_p^R\right]$ space

(b) Versus σ_p^R

Figure 35: As in Figures 23 and 24 but for total cloud cover.