Calibration and emulation of TIE-GCM

Serge Guillas
School of Mathematics
Georgia Institute of Technology

Jonathan Rougier
University of Bristol

Big Thanks to Crystal Linkletter (SFU-SAMSI summer school)

Supported by the Statistical and Applied Mathematical Sciences Institute

Overview

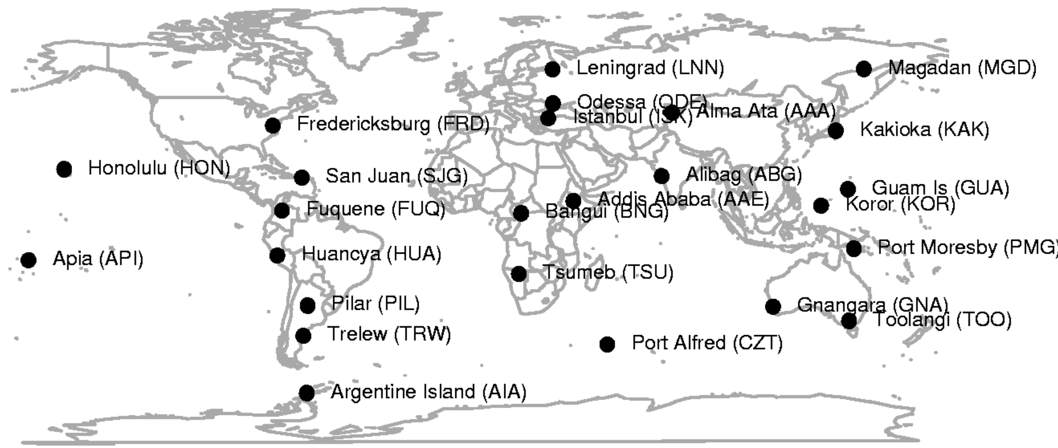
1. Observations
2. Methodology
3. Results

Observations

Ongoing information is being archived at

`http://www.maths.bris.ac.uk/~mazjcr/TIEGCM/`

Location of sites for MAGDDR outputs



Magnetic D-component

1. Amplitude of the migrating tide, unif on [0, 36000]
2. Phase of the tide, periodic, unif on [0, 12]
3. Minimum electron density, log10 unif on [3, 4]

Methodology

<u>Goal of calibration:</u>

Find optimal tuning parameters that give best outputs.

Denote input parameters $z = (x, u)$. Two categories:

- known parameters (controllable parameters x)
- unknown parameters (tuning or calibration parameters u).

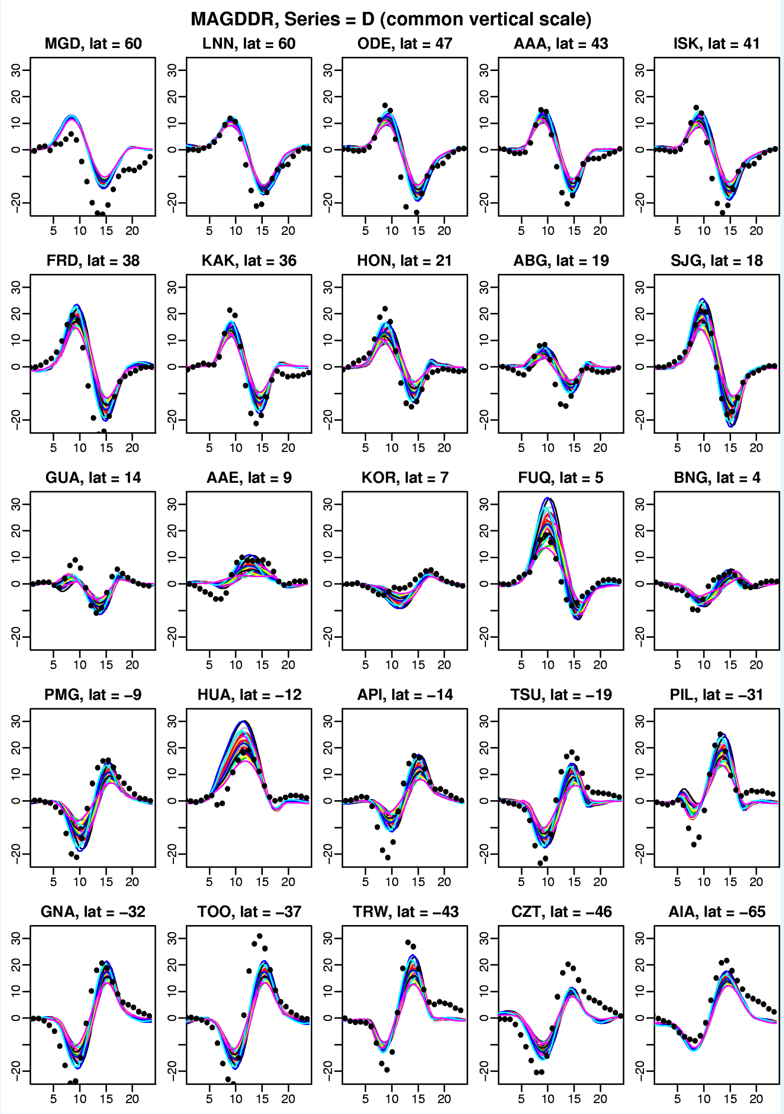
Here:

- controllable parameters x is time (and spatial location)
- calibration parameters u are: amplitude, phase and minimum electron density

- Computer experiments are expensive and time consuming.. Small subset of the calibration parameters: design.
- In between, emulate the computer model by a Gaussian process response-surface (Sacks et al. 1989, Welch et al. 1992, Morris et al. 1993).

Design:

- $y^M(x,u)$ the output of the computer model for the input $z = (x,u)$.
  $y^M(x,u)$ approximation of the reality $y^R(x)$.
- Computer model run at inputs (x,u) in the design D^M.
- Field data collected at inputs x in the design D^F.
- The choice of the design D^M is done through a latin hypercube sampling. This random design is intended to cover as much input space as possible.
- 30 runs of TIE-GCM for the 3 parameters


MAGDDR, Series = D (common vertical scale)
MGD, lat = 60
LNN, lat = 60
ODE, lat = 47
AAA, lat = 43
ISK, lat = 41
FRD, lat = 38
KAK, lat = 36
HON, lat = 21
ABG, lat = 19
SJG, lat = 18
GUA, lat = 14
AAE, lat = 9
KOR, lat = 7
FUQ, lat = 5
BNG, lat = 4
PMG, lat = −9
HUA, lat = −12
API, lat = −14
TSU, lat = −19
PIL, lat = −31
GNA, lat = −32
TOO, lat = −37
TRW, lat = −43
CZT, lat = −46
AIA, lat = −65

Representations of model bias and uncertainty: Kennedy and O'Hagan (2001)

Description of the bias $b_u(x)$ and observation error ε:

$$y^R(x) = y^M(x,u) + b_u(x) \tag{1}$$

$$y^F(x) = y^R(x) + \varepsilon \tag{2}$$

Note:

- We can add a noise ε_η that represents the uncertainty in the code itself (usually small).
- Goal is to find u^* that gives best approximation of reality
- Confounding between u and $b_u(x)$

Gaussian stochastic process (GASP)

- $Z(z), z \in \mathbb{R}^p$ good approximation of the outputs of the computer model.
- GASP unknown function, except at the design points.
- $Z(z)$ is $N(\mu, \sigma^2)$ with correlation function:

$$c(z, z') = exp(-\sum_{k=1}^{p} \beta_k \, |z_k - z'_k|^{\alpha_k}).$$

- The parameters $\mu, \sigma^2, \beta_1, .., \beta_k, \alpha_1, .., \alpha_k$ are constrained (e.g. $1 \leq \alpha_k \leq 2$..here we choose $= 2$).
- Hyperparameters describe the priors on b_u, and $\varepsilon, \varepsilon_\eta$

Estimation of the parameters

For one location: API

Likelihood: We assume i.i.d. normal noises and GASP-induced correlation structures for $y^R(x)$ and $b_u(x)$.

We rescale the parameters on $[0,1]$. We work with log-likelihood.

Estimation

1. Through maximum likelihood (hard!)
2. Bayesian procedure

Here Bayesian procedure,
MCMC approach with Metropolis-Hastings algorithm to draw realizations from the posterior distribution.

Priors

- Uniform for calibration parameters u_1, u_2, u_3.
- Beta for the correlations
- Gamma for the precisions

MCMC iterations: 10,000. Computing time: 5 hours.

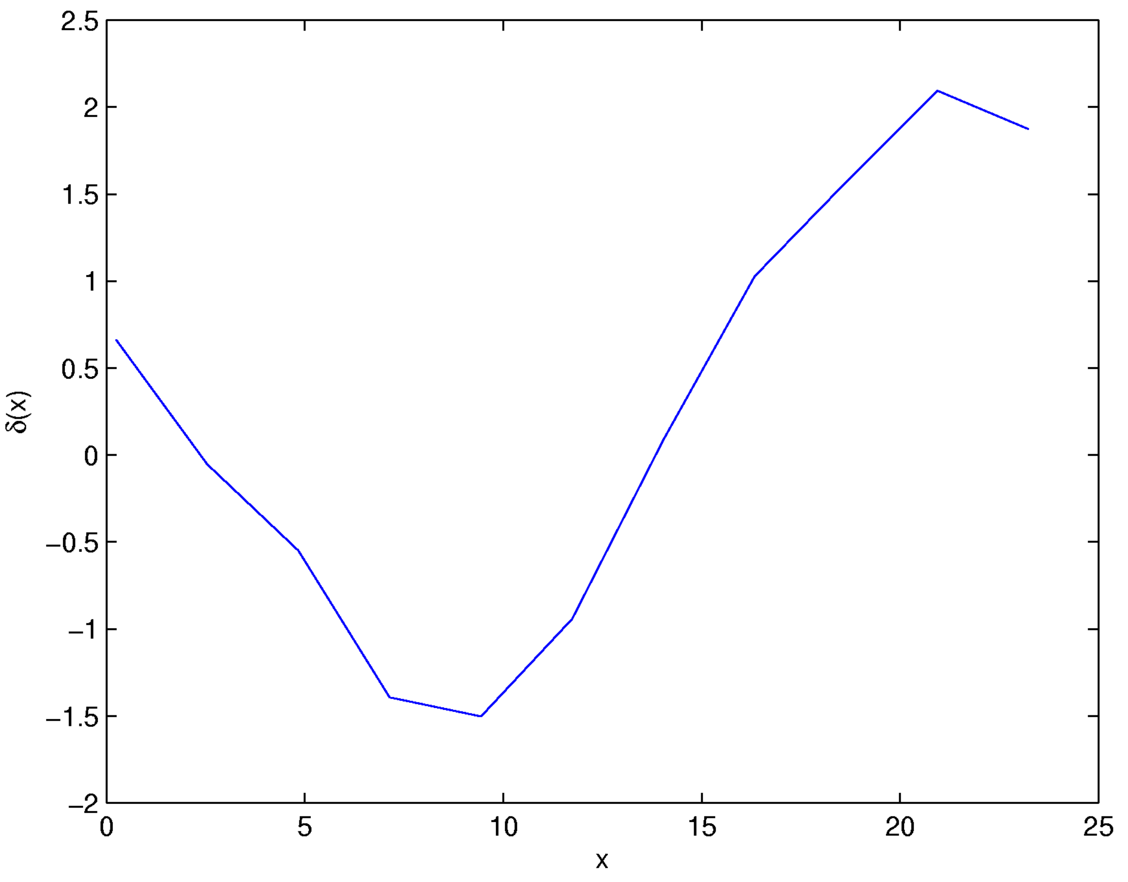
From joint posteriors we get the following:

Results

Emulator and RAQAST outputs

- **Posterior of $y^M(x)$: "emulator"**
- **Posterior of $y^M(x) + b_u(x)$: "prediction"**

Bias



Calibration

Posteriors of u give:

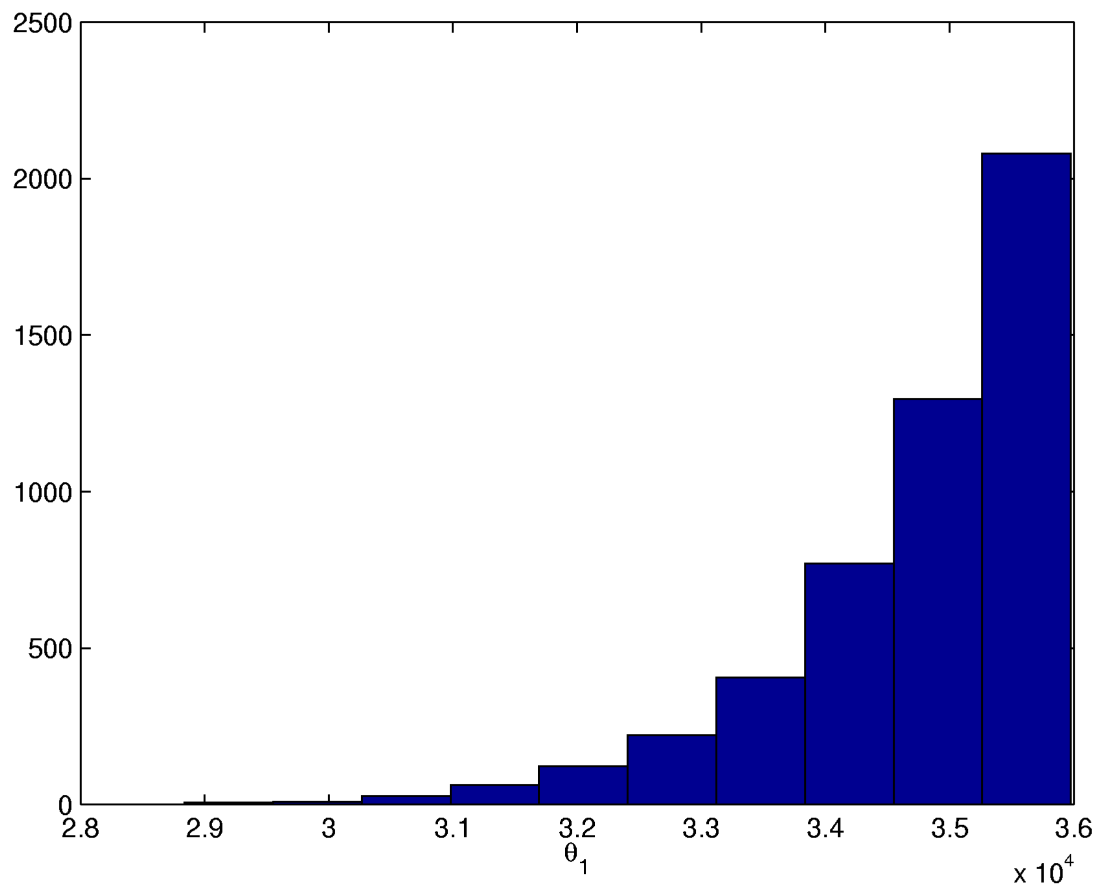


Figure 1: Amplitude

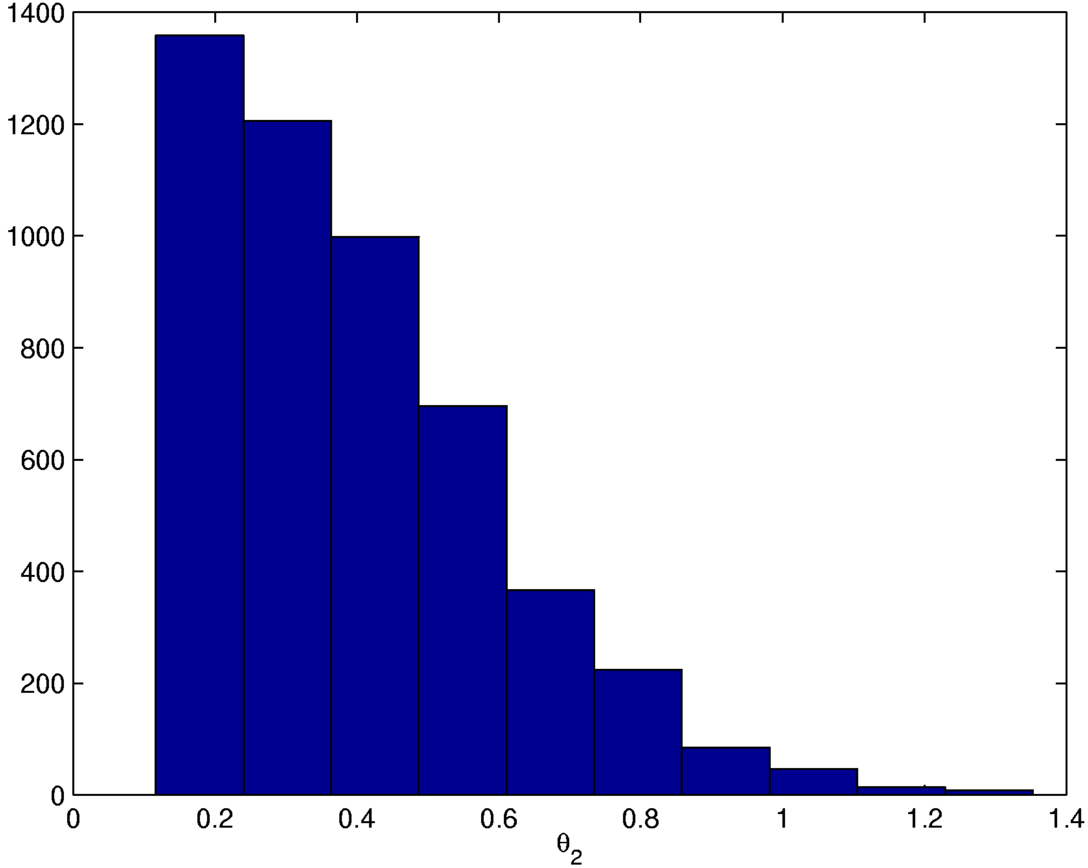


Figure 2: Phase ..wrong!

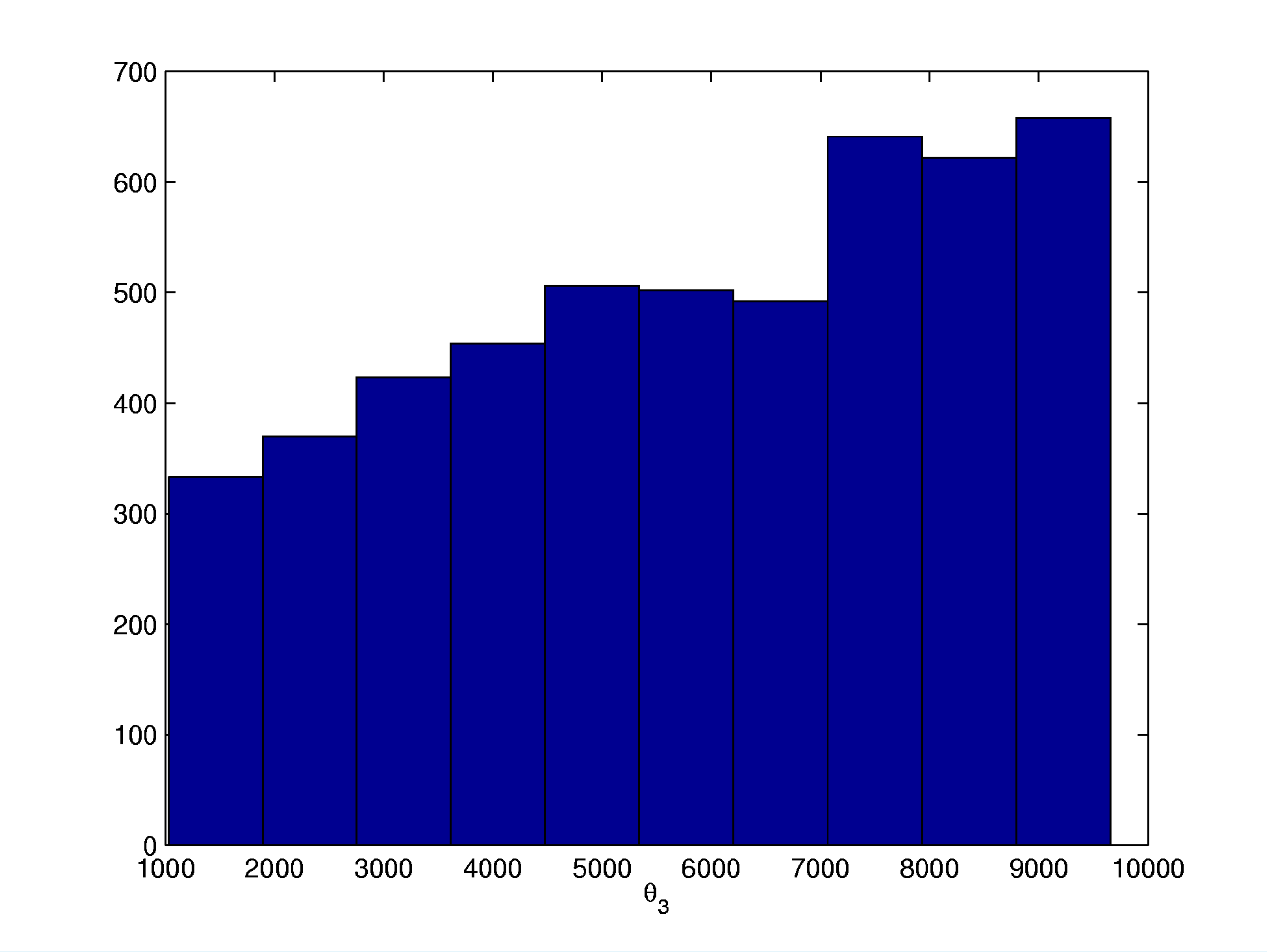


Figure 3: minimum electron density

Functional calibration

Time series:

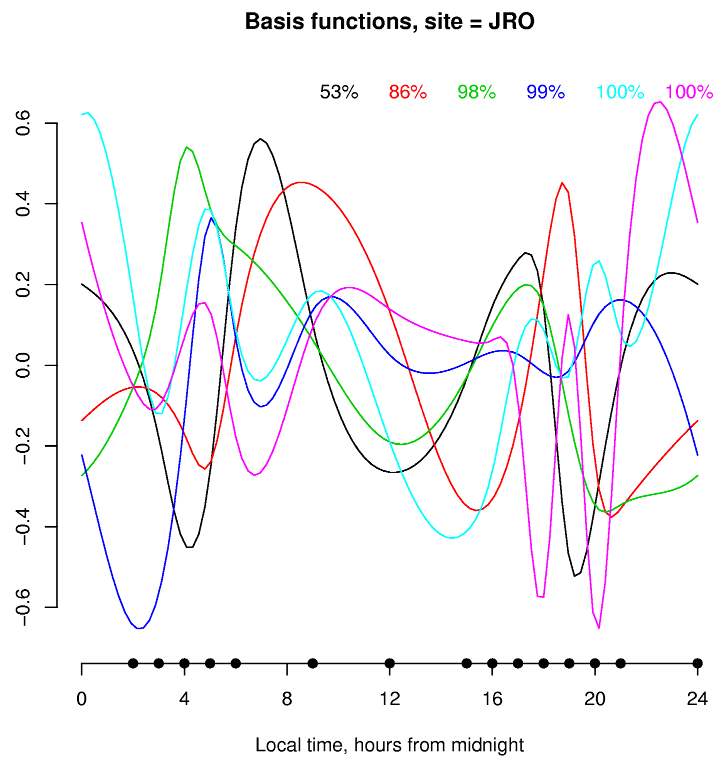
- PCA (Higdon et al., JASA, 2007)
- wavelets (Bayarri et al., Ann. Stat., 2007)

Periodic splines

- For tractability, it is very convenient if the model-evaluations and model-outputs are ‘rectangular’, i.e., they have an outer-product structure where the full set of evaluations can be laid out with rows corresponding to inputs and columns to outputs.
- We ‘regularise’ the model-outputs at each evaluation by fitting them with a periodic spline and then holding the spline at a fixed set of knots on $[0, 24]$. We choose the knots to be dense where we want higher resolution; i.e., around the shoulders.
- We predict the outputs at the knots, and then extend them over the whole of $[0, 24]$ using the periodic spline through the knots (we can quantify uncertainty by sampling).

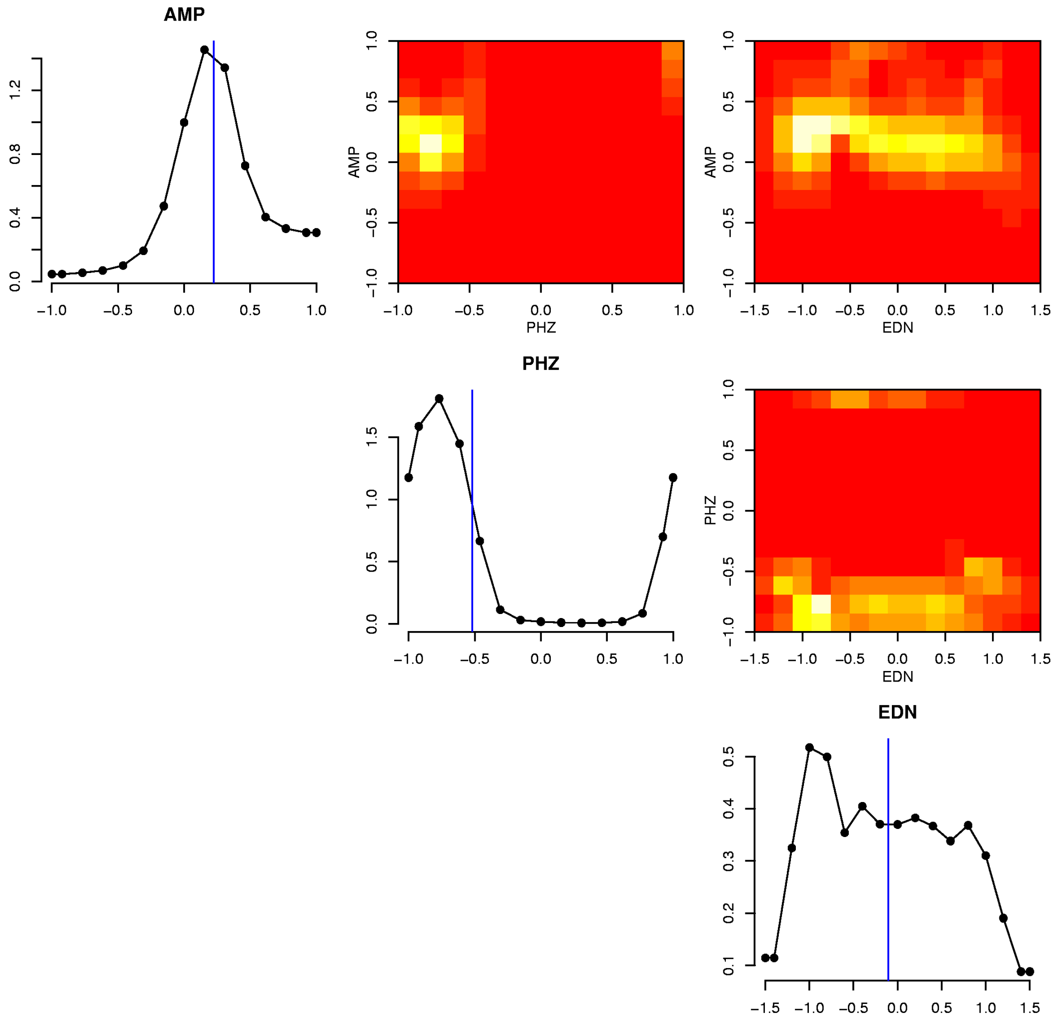
- **For basis functions on $[0, 24]$ we use the right eigenvectors of the SVD of the regularised output matrix (aka Empirical Orthogonal Functions, or EOFs).**

Basis for JRO-drift



Basis functions, site = JRO
53%
86%
98%
99%
100%
100%
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
4
8
12
16
20
24
Local time, hours from midnight

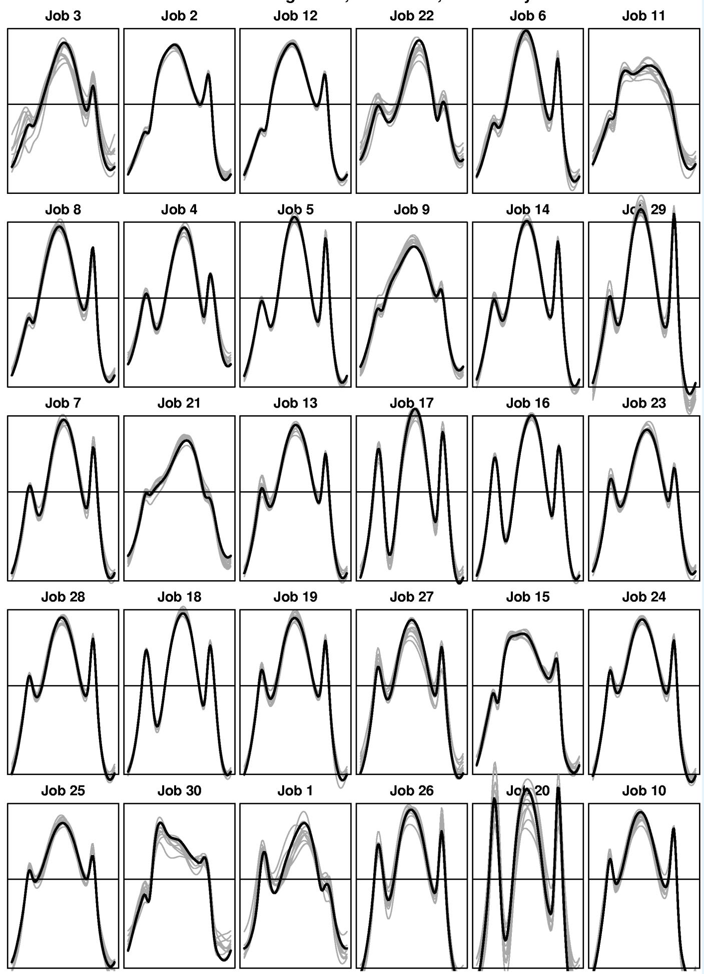
Calibration for JRO-drift


AMP
PHZ
EDN
AMP
PHZ
EDN
AMP
PHZ
EDN

Emulator for JRO-drift

- regressors in time: EOFs
- regression in inputs: Legendre monomials up to second order for AMP and EDN, and two Fourier terms for PHZ
- circular correlation function

Leave−one−out diagnostic, Site = JRO, ordered by EDN



Future plans

- use 7 calibration parameters
- use spatial information
- Modular Bayes for faster computations: use replicates for field uncertainties..only