Supplement of

"Pitfalls and a feasible solution for using KGE as an informal likelihood function in MCMC methods: DREAM$_{(ZS)}$ as an example"

Yan Liu[1], Jaime Fernández-Ortega[2], Matías Mudarra[2], Andreas Hartmann[1,3]

[1] Chair of Hydrological Modeling and Water Resources, University of Freiburg, 79098 Freiburg, Germany.

[2] Department of Geology and Centre of Hydrogeology, University of Málaga (CEHIUMA), 29071 Málaga, Spain

[3] Department of Civil Engineering, University of Bristol, Bristol, UK.

This supplement contains one figure (Figure S1) to demonstrate the model performance during the calibration period for the Case study 2 and one text (Text S1) to show the Matlab code on how to set the probability function using KGE and gamma distribution function.

- **Figure S1**

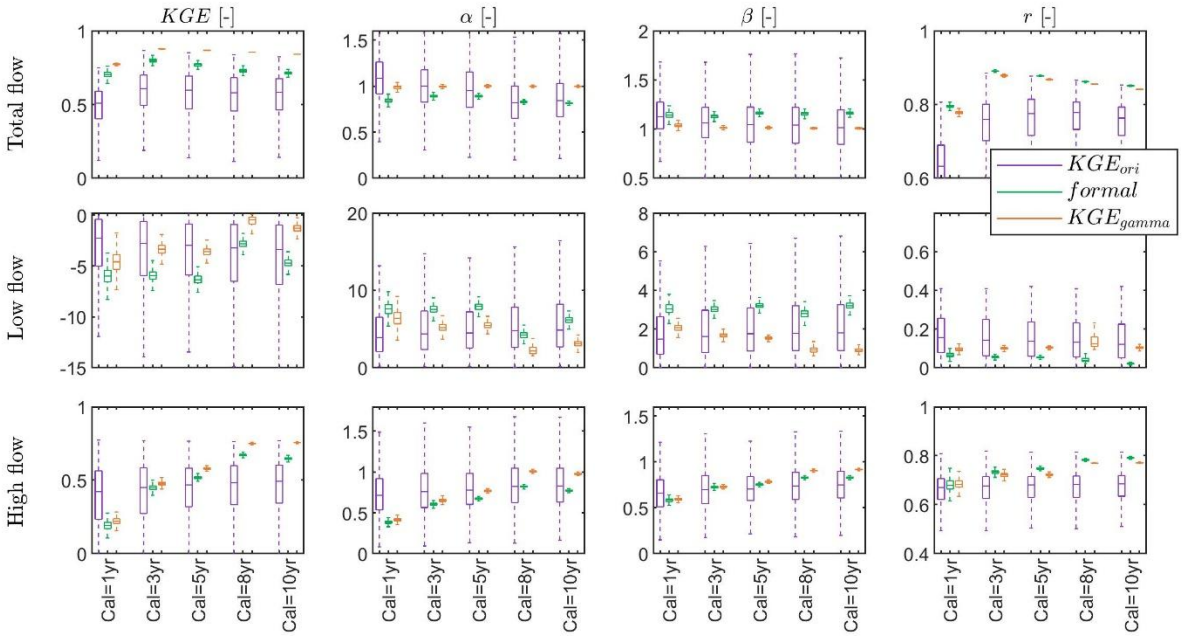

Figure S1 General performance (KGE), variability (α), non-scaled bias (β) and correlation (r) for total flow, low flow (smaller than 10th percentile of observed discharge) and high flow (larger than 90th percentile of observed discharge) during the calibration period using the original KGE (KGE_{ori}) as the likelihood function, the formal likelihood function RMSE, and our approach using KGE and gamma distribution to derive probability density (KGE_{gamma}) with varying amount of observations (1-year to 10-year) in calibration, for instance, calibration with 1-year observations is shown as Cal=1yr. The boxplot shows the performance of the last 25% of all simulations, which is used to approximate the "true" system behavior in DREAM$_{(ZS)}$. The optimal value for KGE, α, β and r is one, and the closer to one the better the performance.

- **Text S1**

Step 1: set the likelihood function type to the log-likelihood

```
% Choose the number 2, which is the log-likelihood
DREAMPar.lik = 2;
```

Step 2: set DREAM(zs) to call the function calculate the log-likelihood with KGE and gamma distribution function

```
% Define name of function (.m file) for posterior exploration
Func_name = 'loglikelihoodCalculator';
```

```
function [log_L] = loglikelihoodCalculator(parameter)
% calculate the log-likelihood using KGE and gamma distribution function
% based on the simulation of a model (such as HBVhumped) and pass to DREAM

% Run the simulation
[Qsim] = HBVlumped(parameter);

% Load observations
Qdata = load('obs.mat');
Qobs  = Qdata.Q;

% calculate KGE
n=length(Qobs);
KGE = KGECompute(Qsim,Qobs);
ED = 1-KGE;

% calculate log-likelihood function
gammapdf = gampdf(ED,0.5,1);
log_L = 0.5*n*log(gammapdf);

end
```

```
function [KGE]=KGECompute(sim,obs)
% compute KGE
obsMu = mean(obs);
simMu = mean(sim);
obsSigma = std(obs);
simSigma = std(sim);
covAllSO = cov(sim,obs);
covSO    = covAllSO(1,2);
r        = covSO/(obsSigma*simSigma);
alpha    = simSigma/obsSigma;
beta     = simMu/obsMu;
KGE      = 1-sqrt((r-1)^2+(alpha-1)^2+(beta-1)^2);
end
```