

*Geophysical Research Letters*

Supporting Information for

**Fresh Look at Variography: Measuring Dependence and Possible Sensitivities Across Geophysical Systems from Any Given Data**

R. Sheikholeslami1,2,3 and S. Razavi2,3,4

1School of Geography and the Environment, University of Oxford, Oxford, UK

2School of Environment and Sustainability, University of Saskatchewan, Saskatoon, Canada

3Global Institute for Water Security, University of Saskatchewan, Saskatoon, Canada

4Department of Civil, Geological, and Environmental Engineering, University of Saskatchewan, Saskatoon, Canada

**Contents of this file**

Text S1 to S4

Figures S1 to S2

Tables S1 to S2

**Introduction**

This file contains supplementary information for the manuscript A Fresh Look at Variography: Measuring Dependence and Possible Sensitivities Across Geophysical Systems from Any Given Data by R. Sheikholeslami and S. Razavi. The file provides details of algorithmic implementation of the proposed method, the utilized hydrologic models, software availability, and additional figures and tables to explain the methodology and numerical experiments.

Text S1: The Relationship Between Variogram and Covariance Function

Consider the model responses at any pair of sample points, and . We begin by estimating their expected squared difference:

To relate this to covariance function , note that if , we define , where is the spatial covariance function of the . For any stationary process with mean , the covariance between and can be calculated as follows:

Consider the case that , then . The same argument in Eq. (2) shows that:

By substituting Eq. (3) and (2) into Eq. (1), we can express the expected squared differences for any pair of sample points in terms of covariance function:

Now, recall the definition of the variogram :

For any stationary process, it can be shown that the variance of the process is equal . Hence, observing from Eq. (5) and (4), we obtain the following simple equation for variogram for all distances **:**

Text S2: Algorithmic Implementation of the D-VARS method

There are three main steps involved in implementing the proposed D-VARS framework: (i) choosing the covariance function, (ii) estimating the hyperparameters of the covariance function, and (iii) computing the VARS-based sensitivity indices. In the following sub-sections, we provide the algorithmic implementation of these three steps in detail.

***Step 1: covariance choice***

As mentioned in Section 2.2 of the main manuscript, we restrict our analysis to a family of covariance functions, wherein the correlation between and is a function of . There are several types of functions that satisfy this condition, such as exponential, linear, Gaussian, spherical, and Matérn functions. A list of stationary covariance functions can be found in Abrahamsen (1994); Chilès and Delfiner (1999); Rasmussen and Williams (2006). While we note that theoretically any stationary function can be used in D-VARS, for the sake of simplicity, we utilized the linear (LIN) correlation functions in this study, as follows:

where are the hyperparameters of the correlation function.

Let us define elements of the correlation matrix for the response variable of interest, given a set of sample points with size , as follows:

Typically, the response of the simulation models exhibits an anisotropic behavior regarding the sensitivities, wherein the variability differs along with different input factors (i.e., sensitivity is variant with respect to direction). This means that different correlations, and as a result different variograms can be thought of in different directions. We accounted for this in correlation functions defined in Eq. (7), because we allow independent parameters in each of the dimensions. The LIN correlation function is illustrated in **Figure S2** for different values of the hyperparameters. In other words, the hyperparameters can be interpreted as an indication of how fast the correlation changes when we move along the *j*-th direction in the input-output space.

There are some algorithmic improvements for the D-VARS that should be considered but were left out of the present work. Future research informed by this study may include implementing various types of correlation functions, such as exponential, Gaussian, and Matérn functions, reflecting prior belief in characteristics of the response surface (e.g., smoothness, multi-modality, etc.). The linear correlation function used in this study has only one parameter, and thus it cannot distinguish other scale-dependent sensitivity indices such as IVARS10 of the original VARS algorithm, which can be seen as the average small-scale sensitivity, while IVARS50 (used in the present work) can be interpreted as the average large-scale sensitivity.

***Step 2: parameter estimation***

After choosing the correlation function, we need to identify its hyperparameters. Maximum likelihood estimation is a popular method that can be effectively employed at this stage to learn the hyperparameters from the data. Ignoring the constant terms, under the Gaussian assumption, the negative log-likelihood function can be written as follows for a set of system responses at all sample points (Sacks et al. 1989; Welch et al., 1992):

where is determinant of the correlation matrix, is an vector of ones, , and

Therefore, determining the optimal values of the hyperparameters , for the correlation function chosen in Step 1, is equivalent to solving the following optimization problem:

To find in Eq. (11), we used sequential quadratic programing optimization algorithm (SQP) combined with a clustering-based multi-start strategy (hereafter refer to as multi-SQP). Th SQP optimization method is a powerful algorithmic tool for the solution of small- and large-scale problems as well as the problems with significant nonlinearities (Boggs and Tolle, 2000). Having a profound theoretical ground, the SQP method is basically a generalization of the Newton's method, which approximates the Hessian of the Lagrangian function using a quasi-Newton updating scheme (Powell, 1978; Schittkowski, 1983).

However, like other gradient-based search techniques, the SQP needs an initial starting point, and accordingly there is a risk of getting stuck in local optima, which can result in lack of robustness. This becomes more crucial when noticing that the optimization problem in Eq. (11) is usually a difficult problem. By way of example, the correlation matrix may be very ill-conditioned or the log-likelihood function near the optimum may be flat in some situations (for more discussion on the properties the optimization problem in Eq. (11) see Lophaven et al. (2002)). To alleviate the problem of local optimum solutions, one promising strategy would be restarting the optimization algorithm from several initial points to ensure convergence to the maximum-likelihood solution (Regis and Shoemaker, 2013; Regis, 2016). Here, we adopted the multi-start strategy of MacDonald et al. (2015) and Butler et al. (2014). This strategy starts by randomly generating several values using a space-filling design and then employs the *k*-means clustering algorithm for careful selection of a set of initial points from these values. Below is a description of the multi-SQP algorithm:

1. Using a space-filling design, generate a sample of hyperparameters from hypercube. Denote this set by .
2. Evaluate the negative log-likelihood function of Eq. (11), , for all values of forming {}.
3. Among {} select values of that corresponds to the best (smallest) values of .
4. Group these values of the hyperparameters into classes using the *k*-means clustering algorithm. Find the centroids of these clusters.
5. Choose three equidistance points along the main diagonal of the search space , i.e., at , , and , where .
6. Evaluate at the three points obtained from Step 5 and choose the best, which has the lowest .
7. Form the set of initial starting points, found in Step 4 and 6.
8. Run the SQP algorithm for each of the starting points.
9. Return the best solution among the solutions encountered by the SQP algorithm and denote it as .

Here, the progressive Latin hypercube sampling (PLHS) strategy (Sheikholeslami and Razavi, 2017) was used to generate the parameter population. A possible improvement of our method might be employing alternative optimization techniques in the process of parameter estimation of the correlation function. This can potentially improve the computational efficiency of implementing D-VARS; however, one needs to keep in mind that, based on our experience, obtaining the (absolutely) best maximum likelihood solution is not necessary.

***Step 3: sensitivity indices calculation***

The final step in the D-VARS algorithm is the numerical estimation of the IVARS sensitivity indices. After learning the covariance function from given data in Step 2, the directional variograms and integrated variograms can be easily computed from equations (7) and (8) of the main manuscript, respectively. We performed a numerical integration to calculate the IVARS sensitivity indices using a simple method, which estimates the integration over an interval by breaking the area down into trapezoids. For a given portioning, , where , we can simplify the integral in equation (8) of the main manuscript to the following form:

Note that the accuracy of the approximation increases as the resolution parameter, , decreases. In the numerical experiments of the present study, IVARS50 was used to assess the factor sensitivity, which means that Eq. (12) is computed for “50% of the parameter range”, respectively. IVARS-50 indicates the overall contribution of a parameter, including its interaction with other parameters, to variability of the output.

**Step 4: *Evaluating robustness of the D-VARS***

In this study, inspired by Sheikholeslami et al. (2019), we defined a measure to assess how robust is our proposed method in terms of the degree of insensitivity to sampling variation. Recall from Step 2 that the multi-SQP algorithm, utilized for parameter estimation of the correlation function, generates starting points for optimizing the maximum likelihood function in Eq. (11). When the best solution is found, there remains solutions that are not the best in terms of the likelihood function but might yield similar parameter importance ranking. Therefore, we counted the number of times out of these solutions that the ranking is equal to the ranking associated with the best solution.

This procedure has been repeated for all 100 replicates of the five computational budget scenarios, i.e., in total we produced ×100 solutions for each computational budget scenario. A GSA algorithm is perfectly robust if independent replicates of the algorithm with different samples converge to identical results.

We note that typical approaches to assess robustness based on bootstrap (Efron, 1982) have limited applicability in some given-data GSA methods, because of ill-conditioning of covariance matrix caused by samples with non-unique members (i.e., when the two or more sample points are identical).

Text S3: Case Studies

We chose two classic hydrologic models, namely HYMOD and HBV-SASK as our testbed, because of their high computational efficiency and that they have already been extensively used in GSA studies (Song et al., 2015). In future work, we will test how the performance of D-VARS will scale to models with varying complexity when the dimensionally of the model increases (see Sheikholeslami et al., 2019; Liu et al., 2020).

Note that, in this paper, we chose to test D-VARS on computer experiments for the following reasons. First, the full properties of the underlying relationships were available through computer models and therefore we could benchmark our results against the ‘true’ dependencies between different variables in question. Second, we could replicate our experiment many times, by resampling from the models with increasing sample sizes, to assess the convergence and robustness of the new method.

1. **The HYMOD model**

We used the HYMOD conceptual rainfall-runoff model as the first case study. This model is based on the probability-distributed soil storage capacity principle and given its simplicity (but robustness), is widely applied in hydrologic modelling community for different purposes. In this study, we adopted the version developed by Vrugt et al. (2003), which was configured for the Leaf River catchment, a 1950 km2 humid catchment located north of Collins, Mississippi, USA.

The HYMOD model simulates daily streamflow time series, and it requires precipitation and potential evapotranspiration as input data. The runoff generation in HYMOD is modeled using a rainfall excess model and its structure has two main components: (i) soil moisture routine controlled by parameters ***bexp*** and ***Cmax***, and (ii) routing module controlled by parameters ***alpha***, ***RS*** and ***Rq***. There are two sets of linear reservoirs in routing module, i.e., three linear reservoirs for the quick flow generation and one linear reservoir for the slow flow generation. For more details on HYMOD see Boyle et al. (2000) and Wagener et al. (2001). The parameters of the HYMOD are listed in **Table S1**.

1. **The HBV-SASK model**

For the second case study, we applied the HBV-SASK conceptual model to evaluate the performance of the proposed D-VARS method. HBV-SASK is adopted from the original HBV model (Bergström, 1995) and was developed by the second author for educational purposes (Razavi et al., 2019) at the University of Saskatchewan. HBV-SASK has three main elements (see Fig. 4), namely (i) snow module, (ii) soil moisture module, and (iii) flow routing module. The latter consists of two (fast and slow) reservoirs for surface runoff (non-linear reservoir) and baseflow (linear reservoir) generations.

The model is forced by the time series of precipitation, temperature, and long-term average potential evapotranspiration, and transforms the simulated flow using a triangular weighting function for channel routing. Here, we applied the HBV-SASK to simulate streamflow of the Oldman River basin, a catchment of 1435 km2 located in western Canada. The basin is mostly forested in its western part, whereas its eastern part is used for intensive agriculture. Most of the runoff in the basin is due to the snowmelt, and thus a significant increase in spring runoff can be observed within this catchment. For more details see Razavi et al. (2019). The 12 parameters of the HBV-SASK is described in **Table S2**.

1. **Synthetic input-output datasets for D-VARS**

In the first case study, we chose the five parameters of HYMOD as the inputs and a goodness-of-fit metric to observations, Nash-Sutcliffe Efficiency (Nash and Sutcliffe, 197), as the output. This case study was designed to represent a widely adopted GSA setting for parameter screening, which can inform the process of calibration (Gupta and Razavi, 2018). The second case study, instead, was made as an example of more modern applications of GSA, where the purpose is learning about the system behavior under uncertainty and non-stationarity (Razavi et al., 2020), regardless of the quality of fit to observations (Razavi and Gupta, 2019; Do and Razavi, 2020). Therefore, the 12 parameters of HBV-SASK were chosen as the inputs, while the output was the model’s estimate of flood peak with the 10-year return period. To compute the 10-year flood peak for each model run (under a different parameter set), we fitted a generalized extreme value (GEV) distribution to annual maximum peak discharges extracted from the simulated streamflow times series over the historical period. This case study might also be viewed as an example for flood frequency analysis in ungauged, where one seeks to know which uncertain parameters control the uncertainty in flood estimates the most. We note that these case studies are for demonstration only, and therefore, decisions like choosing GEV were rather arbitrary.

Text S4: Software Availability

All the computational experiments were performed with the VARS-TOOL software (Razavi et al., 2019; freely available at <http://vars-tool.com/>). VARS-TOOL is a multi-approach GSA toolbox designed for comprehensive sensitivity and uncertainty analysis. The proposed D-VARS method, PLHS sampling strategy, and the case studies are available in this software package.



Figure S1. An illustrative example of a response surface and its corresponding variogram surface in three dimensions. Note that, in VARS framework, variogram, , is the rate of variability of response in the direction of *j*-th factor at the scale . Sensitivity is the rate of variability attributed to the *j*-th factor and is a scale-dependent concept.



Figure S2. Correlation functions for different values of the hyperparameter . Note that the larger the values of , the more rapid are the changes in the response surface even over small distances of , leading to higher variability/sensitivity along the *j*-th factor.

Table S1. Description of the parameters of the HYMOD model and their feasible ranges

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Range** | **Description** |
| *Cmax* | 1.00 ‒500.00 | Maximum storage capacity [mm] |
| *bexp* | 0.10 ‒2.00 | Degree of spatial variability of the soil moisture capacity [unitless] |
| *alpha* | 0.10 ‒0.99 | Factor distributing the flow between two series of reservoirs [unitless] |
| *Rq* | 0.10 ‒0.99 | Residence time of the quick release reservoirs [day] |
| *Rs* | 0.00 ‒0.10 | Residence time of the slow release reservoir [day] |

Table S2. Description of the parameters of the HBV-SASK model and their feasible ranges

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Range** | **Description** |
| *TT* | -4.00-4.00 | Air temperature threshold for melting/freezing and separating rain and snow [°C] |
| *C0* | 0.00-10.00 | Base melt factor [mm/°C per day] |
| *ETF* | 0.00-1.00 | Temperature anomaly correction [1/°C] |
| *LP* | 0.00-1.00 | Limit for PET as a multiplier to FC, i.e., soil moisture below which evaporation becomes supply limited |
| *FC* | 50.00-500.00 | Field capacity of soil, in mm. The maximum amount of water that the soil can retain |
| *beta* | 1.00-3.00 | Shape parameter (exponent) for soil release equation [unitless] |
| *FRAC* | 0.10-0.90 | Fraction of soil release entering fast reservoir [unitless] |
| *K1* | 0.05-1.00 | Fast reservoir coefficient, which determines what proportion of the storage is released per day [unitless] |
| *a* | 1.00-3.00 | Shape parameter (exponent) for fast reservoir equation [unitless] |
| *K2* | 0.00-0.05 | Slow reservoir coefficient which determines what proportion of the storage is released per day [unitless] |
| *UBAS* | 1.00-3.00 | Base of unit hydrograph for watershed routing in day |
| *PM* | 0.50-2.00 | Precipitation multiplier to address uncertainty in precipitation [unitless] |