Geophysical Research Abstracts
Vol. 20, EGU2018-10661, 2018
EGU General Assembly 2018

How to Handle Model Crashes in Global Sensitivity Analysis?

Razi Sheikholeslami, Amin Haghnegahdar, and Saman Razavi
University of Saskatchewan, Global Institute for Water Security, Saskatoon, Canada (amin.haghnegahdar@usask.ca)

Global Sensitivity analysis (GSA) is a powerful tool for deepening our understanding of the Earth and Environmental System Models (EESMs), and subsequently improving their predictions. However, the ever-growing complexity of EESMs will pose many types of software development and implementation issues such as parameter-induced simulation crashes. Crashes are mainly caused by the violation of the numerical stability conditions defined in EESMs. These crashes can be extremely computationally costly during execution of a GSA algorithm because they prevent the completion of GSA, and thus waste the rest of (e.g., thousands) model runs. Modelers commonly address this problem by reducing the feasible ranges of parameters and resuming the analysis in a hope to prevent crashes for the next GSA experiment. However, identifying those parameters that are responsible for crashes is a difficult task since it requires analyzing the model's behavior throughout a high-dimensional parameter space.

In this study, the main goal is finding effective strategies that can be applied to substitute model crashes using available information. To do this, we deem crashes as missing data and consider the model response variables as an incomplete data matrix. Then, three efficient strategies, including median substitution, nearest neighbor substitution, and response surface modeling are used to fill in the gaps (i.e. model crashes). We compare the performance of these strategies across two hydrological modeling case studies using a newly developed variogram-based GSA technique (VARS). Our results indicate that the nearest neighbor substitution and response surface modeling effectively handle the model crashes and produce robust GSA results. We show that how this level of robustness is a function of the ratio of the number of crashes to the total number of model runs.