Environmental Modelling & Software 78 (2016) 54–67

Contents lists available at ScienceDirect

Environmental Modelling & Software

journal homepage: www.elsevier.com/locate/envsoft

Fast mechanism-based emulator of a slow urban hydrodynamic drainage simulator

David Machac [a, b, *], Peter Reichert [a, b], Jörg Rieckermann [a], Carlo Albert [a]

[a] Eawag, Swiss Federal Institute of Aquatic Science and Technology, Department of Systems Analysis, Integrated Assessment and Modelling, 8600, Dübendorf, Switzerland

[b] ETH Zurich, Department of Environmental Systems Science, 8092, Zurich, Switzerland

ARTICLE INFO

Article history:
Received 29 May 2015
Received in revised form
6 December 2015
Accepted 7 December 2015
Available online 7 January 2016

Keywords:
Emulation
Gaussian process
Surrogate modelling
Hydrological models

ABSTRACT

Gaussian process (GP) emulation is a data-driven method that substitutes a slow simulator with a stochastic approximation. It is then typically orders of magnitude faster than the simulator at the costs of introducing interpolation errors. Our approach, the mechanism-based GP emulator, uses knowledge of the simulator mechanisms in addition to the information gained from previous simulator runs, so called design data. In this study, we investigate how the degree of incorporating mechanisms into the design of the GP emulator influences emulation accuracy. Similarly to the previous results, we get a significant gain in accuracy already when using the simplest approximation of the mechanisms by a single linear reservoir. However, in this case, we again considerably improve emulation accuracy when using the next two approximations. This allows us to decreases the required number of design data to achieve a similar accuracy as a non-mechanistic emulator.

1. Introduction

The sophistication of urban drainage simulators increases parallel to the available computational power. Hydrological simulators were considered computationally expensive two decades ago (Stieglitz et al., 1999; Axworthy and Karney, 1999) and they still are today (Dobler and Pappenberger, 2013). They numerically solve (usually large) systems of partial differential equations of surface runoff and water flow in the sewer system. Even though new numerical techniques to handle this task more efficiently emerge constantly (e.g. Dongarra et al., 2014), these efficiency gains are compensated by the demand for more accurate and more detailed hydrological models. Computational speed becomes a limiting factor should we need to run a hydrological model tens of thousands of times, e.g. for the purpose of its calibration, for sensitivity analysis, or for uncertainty propagation. We can circumvent this problem by using a different model, which produces approximately the same results, but is orders of magnitude faster. Such a model is called a *surrogate model*. The use of such a surrogate should always be followed by runs of the original simulator to validate the credibility of the results and, potentially, increase their accuracy.

The simplest techniques to build surrogate models is to build a *lower-fidelity model* of a *high-fidelity simulator*, by simplifying the original model or by reducing the accuracy of its numerical solution. An example of the application of the former concept is given by Vanrolleghem et al. (2005), examples of the latter concept are the *multiscale finite volume technique* developed by Lunati and Jenny (2008) or the simulator by Forrester et al. (2006). A comprehensive overview of model-reducing algorithms used to achieve better performance is given by Gugercin and Antoulas (2004).

Another family of surrogate models are *data-driven surrogates*, where data in the name refers to pairs of inputs/outputs of the full simulator, so-called *design data*. These surrogates are more universal as they do not need to consider the structure of the model underlying the simulator. On the other hand, these surrogates suffer from the *curse of dimensionality* regarding the number of simulator parameters (Bellman, 1956; Asher et al., 2015) by requiring strongly increasing sizes of the set of design data with increasing dimension of the parameter space. Due to the consideration of the model structure, lower-fidelity models do not need any design data and are therefore somewhat less sensitive to this problem.

* Corresponding author. Eawag, Swiss Federal Institute of Aquatic Science and Technology, Department of Systems Analysis, Integrated Assessment and Modelling, 8600, Dübendorf, Switzerland.
E-mail address: david.machac@eawag.ch (D. Machac).

http://dx.doi.org/10.1016/j.envsoft.2015.12.007

Data-driven surrogate modelling encompasses many different, well-known, methods, such as Kriging (Cressie, 1990), artificial neural networks (ANN) (Zurada, 1992) or polynomial chaos expansion (Sudret, 2008). Of these techniques, especially ANN gained some popularity in urban drainage modelling (Giustolisi and Laucelli, 2005).

A comprehensive overview of surrogate modelling techniques, both lower fidelity and data-driven, in hydrology can be found in Razavi et al. (2012). This overview, however, shows that most articles about these methods do not include analytical estimates or measured results of the dependence of computation time on the size of the design data set, model complexity, and output dimension.

A particularly well-known class of data-driven surrogate models, called *Gaussian process (GP) emulators* (Kennedy and O'Hagan, 2001; O'Hagan, 2006), are based on formulating a prior as a Gaussian stochastic process and then conditioning this process on the design data to construct the emulator as the posterior. The emulator is then a statistical approximation of the simulator and provides uncertainty estimates in addition to the best estimate. To build surrogate models for dynamic simulators that produce time-series of results, we need to extend this basic emulation concept. The naïve technique of adding time as another input is not only inefficient, but can also lead to computational issues when the time points are densely spaced. This led to the development of new methods that were specifically designed to emulate dynamic simulators (Conti et al., 2009; Higdon et al., 2008; Bayarri et al., 2007; Bhattacharya, 2007; Castelletti et al., 2012). The idea underlying one of these approaches (Liu and West, 2009), is to formulate a linear state-space model and use a Gaussian process as a function of simulator parameters to represent the noise terms of replications of this model for design data sets. Conditioning this dynamic stochastic model with all design data sets then leads to the dynamic emulator.

Reichert et al. (2011) and Albert (2012) proposed to combine the advantages of low-fidelity surrogate models that use the knowledge of the model structure of the simulator with those of data-driven approaches in the form of GP emulation. Their concept is to consider the mechanistic knowledge about the simulator by formulating a stochastic, linear state-space model as a simplified version of the simulator, formulate the noise term of replicate models as a Gaussian Process in the parameters, and condition the resulting stochastic model to the design data. The idea of combining the two approaches is similar to the *multi-fidelity* surrogate modelling approach Leary et al. (2003); Forrester et al. (2007), but its implementation uses different concepts.

We showed the benefits of considering mechanisms for increasing emulator accuracy for a didactical example of flow through a single channel relying on different numerical approximations of linearized open channel flow equations (Machac et al., 2015). The goal of this paper is to extend these results to an analysis of different spatial simplifications of an urban drainage simulation model (the *Storm Water Management Model, SWMM*) of a real catchment. We focus on emulation of a small number of outputs with respect to the simulator parameters, such as a joint factor to the Manning-Strickler coefficients of the sewer pipes, and not on emulating different input time-series to the modelled system. This is what typically would be required for calibration or sensitivity analysis. Results at the full spatial resolution of the network could then be obtained by running the full simulator with the calibrated parameters. From the conceptual point of view, emulating input time-series could be done similarly, however, this may result in practical difficulties due to the large expansion of the dimensionality of the parameter space. We also analyze the dependence of the gain in simulation time and, in particular, its dependence on the size of the design data set.

2. Case study and urban drainage simulator

The future application we have in mind, for our emulator, is the calibration of the parameters of a SWMM model to a few measured time-series. These time-series are comprised of pairs of rainfall events and measured outflows of a catchment with each of these time-series being several hours long. To test the adequacy of the emulator for such a setting, we use a synthetic rainfall that excites the storage tank and makes the response strongly nonlinear. This means that the conditioning process of the stochastic, linear model on which the emulator is based, will be very important to get a good approximation to of the simulator response by the emulator. This is important for the test of different emulators based on different simplifying models. On the other hand, using the synthetic input means that we do not have observed data. In this paper we will thus focus on quantifying the emulation accuracy as a function of the size of the design data set and the approximating linear model on which the emulator is based.

2.1. Adliswil catchment

We focus on a part of the urban drainage system of the city of Adliswil in the canton of Zurich, Switzerland, spanning an area of 162.8 ha. An overview of the situation is shown in Fig. 1. We investigate the outflow at a wastewater treatment plant (WWTP), which then discharges into the river Sihl. Secondarily, we investigate water depth in a particular sewer manhole located approximately in the middle of the modelled area.

In order to investigate the influence of the topology of the catchment on the emulator, which is explained in detail later, we divide the catchment into two parts. The northern part (area of 93.69 ha) has a mainly pervious surface (grass etc.) whereas the southern part (area of 69.10 ha) is more urban. This results in different response times for each part. The catchment also contains two combined sewer overflows and one storage tank (located in the southern part) with a complicated set of control rules (more than 80).

2.2. SWMM model

Our SWMM model of the Adliswil catchment has 244 subcatchments and 460 conduits and was created from GIS data. The storage tank, which is non-linear due to its shape and control rules, is also modelled, albeit with a simplified set of rules. We generate an artificial rain event, as seen in Fig. 2, which is strong enough to activate these non-linear elements in a short time. A simulation with this rain event of duration of 100 min with a one-minute time step takes approximately 3 s[1] on a CPU Intel Core i7-2600 CPU @ 3.40 GHz.

The model contains thousands of parameters and it is infeasible to emulate with respect to all of them, as we would need to use a large design data set, which would in turn result in a slower emulator (this is explained later). However, a very large amount of parameters would also make calibration of SWMM infeasible due to parameter identifiability problems (Haag, 2006). Instead, we pick a

[1] Although this seems like a short time that does not require an emulator, we have to keep in mind that it is just a test case on a small catchment spanning a short time period. Typically in an engineering practice, the simulation spans much longer periods. The aim of this work is to compare various simplified models and this comparison requires computationally intense parameter estimations, hence the short simulation time of 100 min.