

# Benchmarking the efficiency of a metamodeling-enabled algorithm for the calibration of surface water quality models

K. Kandris, E. Romas  and A. Tzimas 

## ABSTRACT

Computational efficiency is a major obstacle imposed in the automatic calibration of numerical, high-fidelity surface water quality models. To surpass this obstacle, the present work formulated a metamodeling-enabled algorithm for the calibration of surface water quality models and assessed the computational gains from this approach compared to a benchmark alternative (a derivative-free optimization algorithm). A radial basis function was trained over multiple snapshots of the original high-fidelity model to emulate the latter's behavior. This data-driven proxy of the original model was subsequently employed in the automatic calibration of the water quality models of two water reservoirs and, finally, the computational gains over the benchmark alternative were estimated. The benchmark analysis revealed that the metamodeling-enabled optimizer reached a solution with the same quality compared to its benchmark alternative in 20–38% lower process times. Thereby, this work manifests tangible evidence of the potential of metamodeling-enabled strategies and sets out a discussion on how to maximize computational gains deriving from such strategies in surface water quality modeling.

**Key words** | calibration, hydrodynamics, metamodeling, water quality modeling

K. Kandris (corresponding author)

E. Romas 

A. Tzimas 

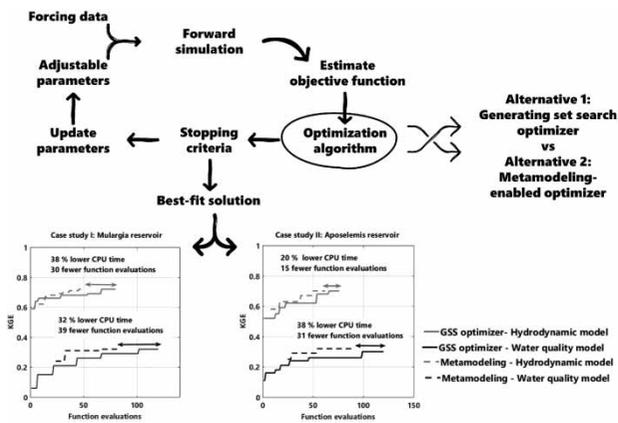
Emvis Consulting Engineers SA,  
Paparrigopoulou 21,  
Agia Paraskevi 153 43,  
Greece

E-mail: [kkandris@emvis.gr](mailto:kkandris@emvis.gr)

## HIGHLIGHTS

- A derivative-free algorithm and a metamodeling-enabled algorithm were formulated for solving inverse surface water quality problems that employ high-fidelity models.
- The algorithms were benchmarked in two surface water reservoirs.
- The algorithms performed robustly in both case studies.
- The metamodeling-enabled algorithm required up to 38% lower process times to solve the inverse problem equally well to the derivative-free algorithm.

## GRAPHICAL ABSTRACT



## INTRODUCTION

Model calibration is essential for the credible prediction of the water quality status of surface waters. As Shimoda & Arhonditsis (2016) postulate, in process-based, water quality models, calibration is typically performed through trial-and-error approaches, during which model parameters are manually fine-tuned until the model confirms the observations. Manual calibration is labor-intensive and entails a high degree of expert opinion and, thus, subjectiveness. The alternative to manual calibration, i.e. automatic calibration, gathers momentum in the literature, but its diffusion within the bibliographic corpus remains limited (Vinçon-Leite & Casenave 2019), partly because the automatic calibration of high-fidelity, process-based models is computationally expensive, especially when models tend to expand in terms of their domain, their numerical resolution and the physical processes they consider. Thus, it is meaningful to devise robust and efficient automatic calibration algorithms that would require tolerable computational budgets.

The present technical note formulated a metamodeling-based approach for the calibration of numerical surface water quality models in search of a computationally efficient automatic calibration framework.

Metamodels are cheaper-to-run models that approximate the dominant features of high-fidelity models. The taxonomy of metamodels comprises (a) data-driven methods

that capture the complex model output over multiple snapshots of the original model, (b) model reduction methods, and (c) multiscale methods, based on the simplification of the underlying physics or the reduction of numerical resolution (Asher et al. 2015). In this work, a data-driven metamodel of the original model was developed to mimic the latter's behavior and, thereby, to reduce the process times required in search of optimal parameter sets.

The computational gains from the utilization of cheaper-to-run metamodels are not self-evident. Despite the optimism that surrounds metamodeling-enabled optimizers, they can be inefficient and unreliable to deal with complex numerical models (Asher et al. 2015). Therefore, the extent to which the metamodeling-enabled optimizer represents an improvement over a derivative-free optimizer was assessed in the calibration of three-dimensional, water quality models of two surface water reservoirs located in Mulargia (Italy) and Aposelemis (Greece).

## METHODS

The calibration process was treated as a black-box optimization problem (it will be referred to as the inverse problem hereinafter), which comprised three major components: (1) the simulation of the original process-based models

(i.e. the forward problem), (2) the estimation of the discrepancy between simulated and observed quantities (i.e. the objective function), and (3) the algorithmic search for the parameter set that minimizes the aforementioned discrepancy within the multi-dimensional parameter space. Two alternatives were considered in this component, (a) a derivative-free optimizer and (b) a metamodeling-enabled optimizer. Ultimately, when both alternatives converged to a solution of the inverse problem, the corresponding process times were estimated and, by that, the computational gains associated with the metamodeling-enabled framework were assessed. The remaining section outlines (a) the specifics of the forward problem, (b) the features of the inverse problem, (c) the algorithms used for the inverse problem, and (d) how the computational gains are assessed. Readers who are not close to the topic of surface water modeling may elect to only skim the subsection entitled 'Formulation of the forward problem' and focus on the remaining subsections.

### Formulation of the forward problem

The aim of water quality modeling was to describe the most salient physical, chemical, and biological features of the two reservoirs. Fundamentally, a credible water quality model is based upon a sound hydrodynamic model. Hydrodynamic modeling reproduces water flow and circulation patterns, mixing and dispersion phenomena, and the spatiotemporal variations of water temperature and density, which constitute the driving mechanisms of the transport of sediment, nutrients, and algal populations. Therefore, each water quality model is coupled with its hydrodynamic counterpart. For each case study, hydrodynamic and water quality models were calibrated individually and, thus, the corresponding computational gains will be reported separately.

### Conceptual design of the coupled hydrodynamic and water quality models

Conceptually, hydrodynamics in both water reservoirs was dictated by two external forcings: (1) inflows (i.e. hydrological forcings from the upstream catchments of the reservoirs) and outflows (i.e. water abstraction rates), and (2) meteorological forcings that comprise wind forcings and heat flux exchanges with the atmosphere. In order to describe the

impact of these forcings on the flow patterns and the thermal behavior of the water reservoirs, the following physical processes were considered: (a) free surface gradients, (b) variable-density flow, (c) horizontal density gradients in the pressure, (d) time-varying inflows and outflows, (e) turbulence-induced mass and momentum fluxes, (f) space- and time-varying shear stresses at the water surface and at the bottom, (g) transport of heat and heat exchange through the free surface, and (h) evaporation.

The conceptual design of the water quality models of the reservoirs comprised (a) sedimentation and resuspension of particulate matter, (b) growth and decay of phytoplankton, (c) light extinction, (d) mineralization processes, (e) nitrification, denitrification, and adsorption kinetics, and (f) reaeration and sediment oxygen demand. The water quality model contains 16 state variables, which are grouped as follows: (a) dissolved oxygen, (b) suspended inorganic matter, (c) inorganic dissolved nutrients, (d) inorganic particulate nutrients, (e) particulate detrital organic matter, (f) dissolved organic matter, and (g) primary producers. Apart from the mechanisms occurring within the water column, the bottom of the water reservoir was treated as an active sediment layer. There, detrital organic matter, algae, and inorganic matter could settle or return to the overlying water through resuspension. Due to the lack of field data, decomposition kinetics within the sediment were ignored; the sediment layer did not function as a nutrient source. Only a surrogate sediment oxygen demand was considered to reproduce the observed vertical dissolved oxygen profiles in the reservoirs.

### Numerical implementation

The Delft3D modeling suite formulated the above-mentioned mechanisms into a numerical model.

The hydrodynamic module of the Delft3D modeling suite, i.e. Delft3D-FLOW, was implemented to simulate the three-dimensional unsteady flow and transport phenomena of the reservoirs. The numerical hydrodynamic modeling system Delft3D-FLOW solved the Navier Stokes equations for an incompressible fluid, under the shallow water and the Boussinesq assumptions. The system of equations consisted of the horizontal equations of motion, the continuity equation, and the transport equation for

heat. For the time integration of the hydrodynamic model, Delft3D-FLOW introduces the Alternating Direction Implicit method. A flooding-scheme was selected for the spatial discretization of the horizontal advection terms, while for the vertical advection term, a second-order central difference scheme was applied. To safeguard against non-physical oscillations, which may occur due to the approximation of vertical viscosity terms, a filtering technique, namely a Forester filter, was applied.

The water quality module of the Delft3D modeling suite, i.e. DELWAQ, utilized the hydrodynamic conditions (velocities, water elevations, density, vertical eddy viscosity, and diffusivity) calculated by the Delft3D-FLOW module and solved the advection–diffusion–reaction equation for the model substances described previously. A local-theta flux-corrected transport scheme was adopted for the discretization of transport in the water quality model. This numerical scheme treats horizontal transport using central differences (a second-order method), and if local maxima or minima occur numerical dispersion is added to avoid the occurrence of such extremes. More details in the numerical aspects of the Delft3D modeling suite are provided in *Delft Hydraulics* (2014a, 2014b).

### Model schematization

A fine-resolution rectangular mesh of grid cells was designed to obtain a sufficient representation of the reservoir geometries. In search of a trade-off between accuracy and computational burden, 6,340 active horizontal grid cells of  $50 \times 50 \text{ m}^2$  reproduced the geometry of Aposelemis reservoir, which is a small-yet-deep reservoir, occupying  $27 \text{ hm}^3$ , covering a surface of  $2.0 \text{ km}^2$  with a maximum depth of 41.0 m. Likewise, a rectangular mesh of 33,712 active horizontal cells represented the geometry of Mulargia reservoir, which is a large and deep reservoir, covering an area of  $12.5 \text{ km}^2$ , with a maximum depth of approximately 99 m and a maximum volume of about  $347 \text{ hm}^3$ . Grid coordinates were defined in a Cartesian coordinate system which was adequate to reproduce the relatively simple shape of the reservoirs.

Water column depths were represented by non-uniform  $\sigma$ -grids; a 10-layer grid was considered for Aposelemis and a 15-layer grid was formulated for Mulargia. The relative thickness of the layers allowed for more resolution in the

near-surface area (critical for describing wind-driven flow and heat exchange with the atmosphere) and the mid-depth area (critical for capturing vertical stratification). The built-in anti-creep correction was activated to suppress artificial vertical diffusion and flow in the applied  $\sigma$ -grid model.

Due to limitations imposed for stability and accuracy of the time integration of the shallow water equations in Delft3D-FLOW, the time step used for the hydrodynamic simulations was equal to 30 s. The water quality model was less demanding and, thus, a 15-min time step was selected. Approximately 2.5 and 14 CPU hours of parallel processing were required for a 1-year forward hydrodynamic simulation on a quad-core, 3.4 GHz Intel® i5-7500 processor for Aposelemis and Mulargia, respectively. Nearly 20 and 50 min of CPU time were consumed for the corresponding water quality simulations.

### Formulation of the inverse problem

Model calibration was treated as a box-constrained minimization problem, described as follows:

$$\min f(x) \quad (1)$$

subject to

$$x_{\min} \leq x_k \leq x_{\max}, \quad k \in \{1, \dots, d\} \quad (2)$$

where  $x$  is the  $d$ -dimensional vector of model parameters bound by  $x_{\max}$  and  $x_{\min}$  (note that  $d$  differs for the hydrodynamic and water quality components of the models), and  $f(x)$  is the Kling–Gupta Efficiency (Gupta et al. 2009):

$$\text{KGE} = 1 - \sqrt{(\text{cc} - 1)^2 + (a - 1)^2 + (b - 1)^2} \quad (3)$$

where  $\text{cc}$  is the linear cross-correlation coefficient between observed and modeled values,  $a$  is equal to the standard deviation of modeled values over the standard deviation of the observed, and  $b$  is the mean of modeled over the mean of observed. KGE allows for a multi-objective perspective by focusing on separately minimizing the timing ( $\text{cc}$ ), variability ( $a$ ), and bias of errors ( $b$ ). Ideally, for a perfect model KGE is 1.

Modeled and observed data were compared in three water quality attributes. Hydrodynamic models were calibrated over satellite-derived and *in situ* observed water temperatures, while water quality models were calibrated over satellite-derived and *in situ* observed chlorophyll-*a* concentrations and turbidities. One-year-long calibration periods were considered for both case studies, i.e. year 2016 for Aposelemis and year 2015 for Mulargia.

The dimension of the parameter vector,  $d$ , resulted from a judicious selection of model parameters. Consequently,  $d$  was equal to four for hydrodynamic models (i.e. wind drag coefficient, background horizontal eddy viscosities and diffusivities) and equal to 11 for the water quality models (parameters related to the growth of algae species, light intensity of the water column, and the settling of particulate matter). Parameter boundaries, i.e.  $x_{\max}$  and  $x_{\min}$ , were set by the range of values reported in the literature; parameter boundaries are available in Section 1 of the Supplementary Material.

## Benchmarking alternatives for model calibration

### Alternative 1: a derivative-free optimizer

The first alternative is a derivative-free, generating set search (GSS) method; the *patternsearch* function of Matlab<sup>®</sup> was used. The GSS method evaluates the objective function at a mesh of candidate points, which form a stencil around each iterate. The mesh is scaled along the search directions by multiplying the pattern vectors by constants proportional to the logarithms of the absolute values of components of the current iterate. The first search direction at each iteration is the direction in which the algorithm found the best point at the previous iteration. As soon as the algorithm pools a candidate point that has a lower function value, it is considered as the new iterate, the centre of the stencil is shifted to this new point and the size of the stencil becomes four-fold greater. Otherwise, the algorithm does not change the current point and the size of the stencil becomes four-fold smaller in search of a new iterate. The algorithm stops when: (a) the mesh size is less than  $10^{-3}$  (mesh tolerance criterion), (b) the number of iterations reaches an arbitrarily selected total of 200 iterations (maximum iterations criterion), (c) after a successful poll, the distance between the point found in the previous two iterations and

the mesh size are both less than  $10^{-3}$  (step tolerance criterion), (d) after a successful poll, the change in the objective function in the previous two iterations and the mesh size is less than  $10^{-3}$  (function tolerance criterion).

### Alternative 2: a metamodeling-enabled optimizer

A metamodeling-based, adaptive-recursive framework was developed in Matlab<sup>®</sup> to calibrate the coupled hydrodynamic and water quality models. This framework was realized in six steps (Figure 1).

At Step 1, a set of numerical experiments was designed (i.e. a design of experiments, DoE) to portray and empirically capture the behavior of the underlying system over the range of the  $d$  adjustable parameters (i.e. the design space). According to Razavi et al. (2012), full factorial design, fractional factorial design, central composite design, Latin hypercube sampling (LHS), and symmetric LHS are the most widely tested space-filling methods for DoE in the metamodeling literature. In this work, the LHS method was tested for the initial DoEs. The size of the initial DoE was equal to  $2(d + 1)$ , after Regis & Shoemaker (2007). The size and the space-filling strategies for the DoE are relevant for the efficiency of metamodeling-enabled optimizers (Razavi et al. 2012), but the assessment of their relevance is not within the scope of this technical note.

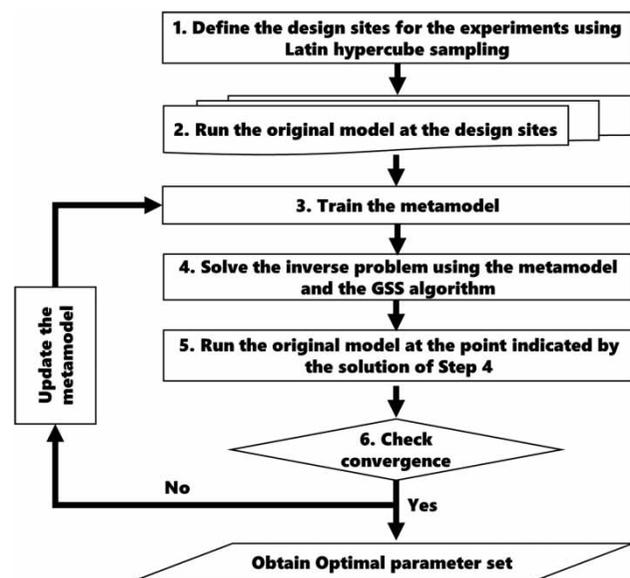


Figure 1 | Schematic overview of the adaptive-recursive metamodeling framework.

Step 2 evaluates the original, high-fidelity model at the design sites generated at Step 1.

Step 3 trains a metamodel on the data produced by the two previous steps. A large variety of data-driven, approximation functions has been tested as metamodels in the literature. Approximation functions vary from simple polynomials, radial basis functions (RBFs) or kriging to more complex artificial neural networks (Razavi et al. 2012). Testing how different approximation functions emulate the original model is beyond the aims of the present work. Therefore, a cubic spline RBF was selected herein, similar to the work of Mugunthan et al. (2005).

Step 4 employs the GSS optimizer (i.e. the first alternative of the benchmark analysis) to solve the inverse problem described in Equations (1) and (2) using the developed metamodel instead of the computationally expensive high-fidelity model. The solution of the inverse problem indicates at which point should the original model be evaluated next.

Step 5 evaluates the original model at the point indicated by the metamodeling-enabled optimizer at Step 4. Then, the KGE value is evaluated using the original model and it is compared to the KGE value predicted by the metamodel at Step 4.

Step 6 checks if (a) the deviation of the KGE values was below a threshold value of 1% or (b) the maximum number of allowed function evaluations has been reached (200 evaluations, as in Alternative 1). If any of the above applies, the optimization process is terminated, and the optimal parameter set is estimated. Alternatively, the current iterate becomes part of the training data set for the metamodel, which is subsequently updated, and Steps 4–6 are re-iterated until one of the above-mentioned stopping criteria is met.

### Assessment of computational savings

Computational saving,  $C_s$ , was estimated based on the relative difference of the process time required for the metamodeling-enabled analysis to reach a solution with the same quality as the GSS method (i.e. KGEs differ by less than 1%):

$$C_s = 100\% \frac{t - t_s}{t} \quad (4)$$

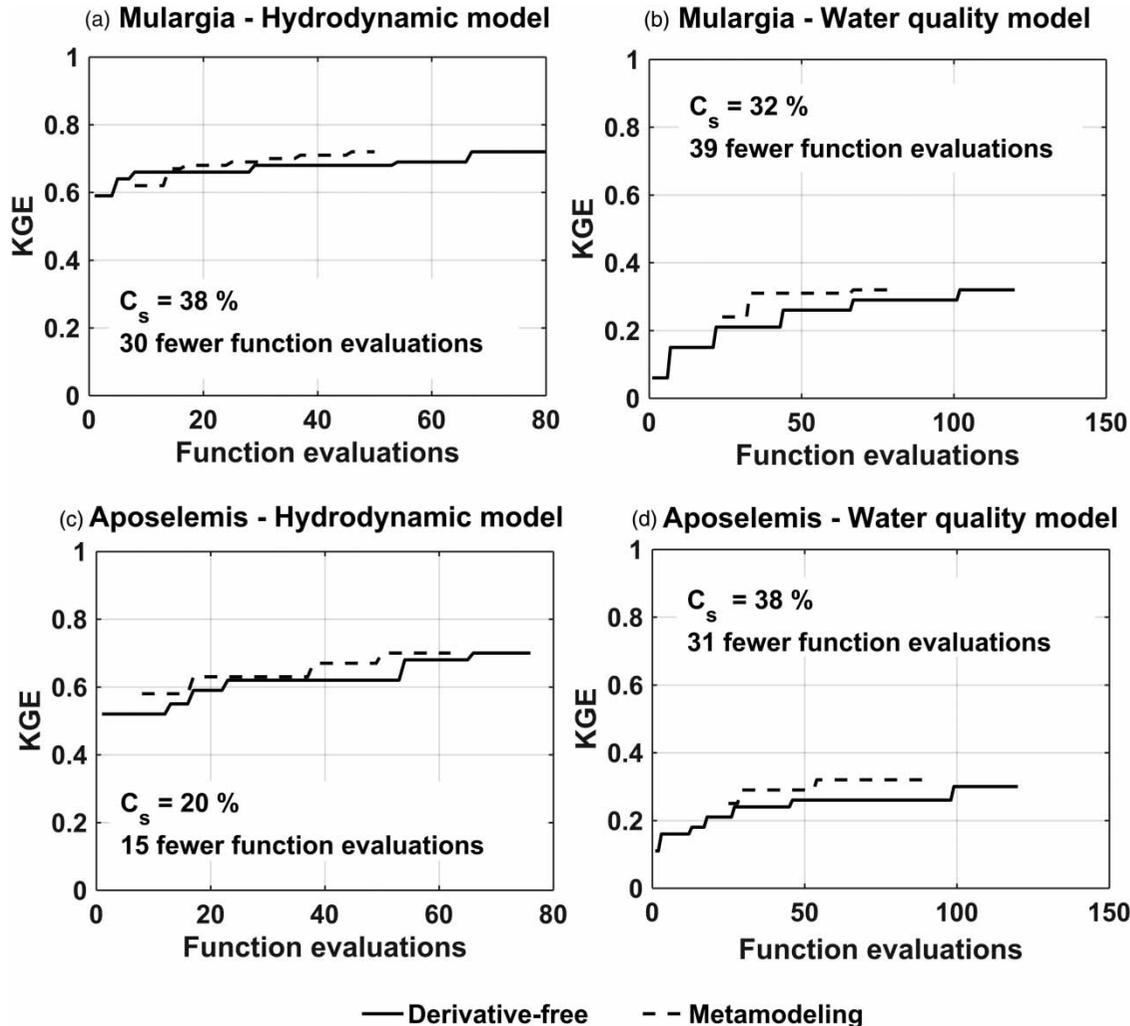
where  $t$  is the process time required to reach a solution through the GSS method (Alternative 1), and  $t_s$  is the process time of the metamodeling-enabled algorithm (Alternative 2). Computational savings were expressed in dimensionless, relative terms to dissociate them from the computational features of the inverse problem (hardware specifics, parallelization options, etc.). Computational savings were estimated for each case study for the hydrodynamic and the water quality models separately.

As Razavi et al. (2012) postulate, in metamodeling-enabled analyses, the process time,  $t_s$ , is divided into three parts: (1) the time required to run the original model ('original model runtime', Steps 2 and 5), (2) the time required to develop, run, optimize, and update the metamodel ('metamodeling time', i.e. Steps 1, 3, and 4), and (3) the time that the analyst needs to identify and create an appropriate metamodeling-enabled analysis framework ('analyst time'). The first two components of the process time, i.e. the original model runtime and the metamodeling time, were considered in Equation (4) when assessing the computational efficiency of the metamodeling-enabled analysis. The analyst time was deemed subjective and, thus, it was neglected.

## RESULTS AND DISCUSSION

The present section benchmarks the performance of the metamodeling-enabled optimizer against the derivative-free algorithm in terms of how well and how efficiently they solved the inverse problem (i.e. the achieved KGE value of the optimum point and the computational time required until the convergence to that optimum). Thus, what follows is a comparative analysis of two algorithms rather than a discussion on how simulated data compared to the observed. However, readers interested in a detailed overview of the outcome of the inverse problem in both case studies should refer to Section 2 of the Supplementary Material.

For the hydrodynamic model, the two alternative algorithms tested herein converged to nearly coinciding optimum solutions, i.e. similar points in the parameter space providing nearly identical values of the objective function, KGE (see Figure 2(a) and 2(c)). The hydrodynamic models reproduced water temperature of the reservoirs with fair accuracy. Specifically, KGE for water temperature was



**Figure 2** | Comparative evaluation of KGE values versus function evaluations for the hydrodynamic and water quality models of Mulargia and Aposelemis for the two alternative optimizers. Please note that for the metamodeling-enabled optimizer the function evaluations for the initial design of experiments have been omitted.

greater than 0.70 for both reservoirs. The decomposed terms of KGE, noted as  $cc$ ,  $a$  and  $b$  in Equation (3), corroborate that prediction errors for water temperature were reasonable in terms of timing, variability, and magnitude: for the two algorithms in both case studies,  $cc$  was in between 0.82 and 0.85,  $a$  ranged from 1.12 to 1.14, and  $b$  varied from 1.13 to 1.15.

With respect to the water quality models, the two calibration algorithms tested herein converged in two distinctive points of the parameter space (see Section 2 of the Supplementary Material for the optimal parameter sets) providing, however, the same quality of fit (compare KGE values in Figure 2(b) and 2(d)). The decomposed

terms of KGE revealed that, in both cases and for both alternative optimizers, water quality models reproduced adequately the variability (term  $a$  of the KGE varied from 1.21 to 1.26) and the magnitude (term  $b$  of the KGE was in the range of 0.75–0.78) of chlorophyll- $a$  and turbidity in the reservoirs, but also delineated the difficulty of capturing the temporal fluctuations of the two quantities of interest (term  $cc$  of the KGE was nearly 0.40 in all cases). This difficulty explains why the predictive skill of the water quality components deteriorated compared to their hydrodynamic counterparts. Nonetheless, model performance is acceptable when compared to similar works reported in the literature. For example, Kuo *et al.* (2006) reported mean

absolute errors of chlorophyll-*a* up to 6.4  $\mu\text{g/l}$ , whereas the corresponding errors in this work were 1.5  $\mu\text{g/l}$  for Aposelemis and 3.1  $\mu\text{g/l}$  for Mulargia. Likewise, Jin *et al.* (2007) reported that the normalized root mean square deviation of their models ranged from 25 to 63%, while herein the corresponding metric ranged from 37% (for Aposelemis) to 40% (for Mulargia).

The two optimizers converged to nearly coinciding KGE values after different process times. For the hydrodynamic models (Figure 2(a) and 2(c)),  $C_s$  from the metamodeling-enabled approach was 38 and 20% for the Mulargia and Aposelemis case studies, respectively (i.e. 30 and 15 fewer function evaluations). For the water quality models (Figure 2(b) and 2(d)),  $C_s$  was 32 and 38% for the two case studies (i.e. 39 and 31 fewer function evaluations). These  $C_s$  values translate in 330 and 68.5 h of CPU time gained for Mulargia and Aposelemis, respectively. These gains are attributed almost entirely to the reduction of the original model runtime, i.e. the fewer forward simulations required for the metamodeling-enabled framework. The metamodeling time was a minor part of the calibration process and varied from 1.0 to 2.4% of the overall process times (data not shown).

Few snapshots of the original models were adequate to train reliable metamodels. For the hydrodynamic models, 50 and 62 evaluations of the original model were sufficient for Mulargia and Aposelemis, respectively. For the water quality models, 82 and 89 snapshots of the high-fidelity models sufficed to capture their behavior for Mulargia and Aposelemis. Nonetheless, generalizations stemming from this finding should be handled with caution as relatively few parameters were considered adjustable herein; for higher-dimensional problems (up to 50-dimensional problems have been solved according to Razavi *et al.* (2012)), it is questionable whether this number of snapshots would suffice.

The present benchmark analysis reports reasonable gains compared to metamodeling-enabled efforts reported in the literature review of Razavi *et al.* (2012). For example, Zhang *et al.* (2009) reported efficiency gains in the range of 20–42% for a comparable number of parameters in the calibration of the Soil and Water Assessment Tool in two watersheds. Likewise, Regis & Shoemaker (2009) reported a 97% reduction in complex model evaluations during the automatic calibration of a groundwater bioremediation model. Thus, the preliminary effort performed herein

underscores the potential for significant efficiency gains through the utilization of data-driven metamodels in surface water modeling. The black-box nature of metamodeling facilitates testing it in diverse surface water systems (regardless of the numerical code used to simulate them) to further investigate its convergence behavior.

## CONCLUSIONS

The considerable growth of computational power over the years has not revoked the computational limitations in the automatic calibration of high-fidelity, process-based simulation models. Metamodeling is an advance in the automatic calibration of such models. Nonetheless, metamodels should not be considered as an *a priori* efficient strategy. Therefore, the present technical note benchmarked a metamodeling-based approach against a derivative-free optimizer in the automatic calibration of surface water quality models of two reservoirs.

Both benchmarking alternatives performed robustly in the solution of the inverse problems of the two case studies. The two algorithms offered solutions that reproduced sufficiently the spatiotemporal dynamics of water temperature and algae growth in the surface water reservoirs.

Benchmarking revealed that the metamodeling-enabled approach is a promising technique for the efficient fine-tuning of parameters of process-based models, in which the conceptualization and parameterization cannot be further simplified without significant loss of fidelity. A computationally workable number of numerical experiments was sufficient to train a radial basis function in reproducing the responses of the high-fidelity model. The computational cost for the development of a cheaper-to-run metamodel was low and was compensated by the computational gains resulting from the fewer evaluations required from the optimization algorithm to reach a solution of the same quality as the solution reached by a derivative-free optimizer.

The benchmarking analysis aspires to initiate a discussion on the prospects, the limitations, and the alternative approaches of metamodeling in water quality models of complex water bodies. First, this work recognizes pathways to optimize efficiency gains, either through employing different approximation techniques or through the design and

analysis of more comprehensive numerical experiments. Second, this work highlights the need to test this framework in higher dimension problems or against different metamodeling techniques, as for example the utilization of multi-fidelity models. All these questions are relevant not only in model calibration, but in a wide range of metamodeling applications spanning from sensitivity analysis to uncertainty estimation.

## ACKNOWLEDGEMENTS

This work was part of the SPACE-O project which has received funding from EU H2020 Research & Innovation Programme under GA No. 730005. The authors kindly thank Dr Pechlivanidis from the Swedish Meteorological and Hydrological Institute for providing simulated hydrological data for the case studies, Ms Schenk from EOMAP GmbH & Co. KG for delivering satellite-derived water quality data for both case studies, and Maria-Antonietta Dessena from ENAS and Stylianos Gyparakis from OAK S.A. for kindly providing the available *in situ* data on water quality for Mulargia and Aposelemis, respectively.

## DATA AVAILABILITY STATEMENT

All relevant data are included in the paper or its Supplementary Information.

## REFERENCES

- Asher, M. J., Croke, B. F., Jakeman, A. J. & Peeters, L. J. 2015 [A review of surrogate models and their application to groundwater modeling](#). *Water Resources Research* **51** (8), 5957–5973.
- Delft Hydraulics 2014a *Delft3D-FLOW User Manual*. WL| Delft Hydraulics (Deltares), Delft.
- Delft Hydraulics 2014b *Delft3D-WAQ User Manual*. WL| Delft Hydraulics (Deltares), Delft.
- Gupta, H. V., Kling, H., Yilmaz, K. K. & Martinez, G. F. 2009 [Decomposition of the mean squared error and NSE performance criteria: implications for improving hydrological modelling](#). *Journal of Hydrology* **377** (1–2), 80–91.
- Jin, K. R., Ji, Z. G. & James, R. T. 2007 [Three-dimensional water quality and SAV modeling of a large shallow lake](#). *Journal of Great Lakes Research* **33** (1), 28–45.
- Kuo, J. T., Lung, W. S., Yang, C. P., Liu, W. C., Yang, M. D. & Tang, T. S. 2006 [Eutrophication modelling of reservoirs in Taiwan](#). *Environmental Modelling & Software* **21** (6), 829–844.
- Mugunthan, P., Shoemaker, C. A. & Regis, R. G. 2005 [Comparison of function approximation, heuristic, and derivative-based methods for automatic calibration of computationally expensive groundwater bioremediation models](#). *Water Resources Research* **41** (11), W11427.
- Razavi, S., Tolson, B. A. & Burn, D. H. 2012 [Review of surrogate modeling in water resources](#). *Water Resources Research* **48** (7), W07401.
- Regis, R. G. & Shoemaker, C. A. 2007 [A stochastic radial basis function method for the global optimization of expensive functions](#). *INFORMS Journal on Computing* **19** (4), 497–509.
- Regis, R. G. & Shoemaker, C. A. 2009 [Parallel stochastic global optimization using radial basis functions](#). *INFORMS Journal on Computing* **21** (3), 411–426.
- Shimoda, Y. & Arhonditsis, G. B. 2016 [Phytoplankton functional type modelling: running before we can walk? A critical evaluation of the current state of knowledge](#). *Ecological Modelling* **320**, 29–43.
- Vinçon-Leite, B. & Casenave, C. 2019 [Modelling eutrophication in lake ecosystems: a review](#). *Science of the Total Environment* **651**, 2985–3001.
- Zhang, X., Srinivasan, R. & Van Liew, M. 2009 [Approximating SWAT model using artificial neural network and support vector machine 1](#). *JAWRA Journal of the American Water Resources Association* **45** (2), 460–474.

First received 18 February 2020; accepted in revised form 21 August 2020. Available online 8 October 2020