Application of High Performance Computing and Efficient Optimization Algorithms for Calibration of Computationally Expensive Water Resources Models

TAIMOOR AKHTAR (1), XIA WEI (2), JINGJIE ZHANG (1)(2) AND CHRISTINE ANN SHOEMAKER (1)(2)(3)

(1): NUS Environmental Research Institute, National University of Singapore, erita@nus.edu.sg
(2): Department of Civil & Environmental Engineering, National University of Singapore
(3): Department of Industrial and Systems Engineering, National University of Singapore

1. Introduction

High performance computing and efficient optimization can be very effective in automatic calibration of computationally expensive water resources models. This paper presents a class of single and multi-objective (MO) parallel (with and without surrogates) global optimization methods, developed in Python, for efficient automatic calibration of water resources models. The search based algorithms are developed in a parallel framework and incorporate non-domination and surrogate models to choose and evaluate many points simultaneously via expensive simulation. The single-objective optimization algorithm is called PR-DDS while the MO algorithm is called MOPLS. The algorithms are applied to calibrate two Soil and Water Assessment Tool (SWAT) based water models (Cannonsville and Townbrook). PR-DDS is applied to the Townbrook model with 1, 8, 16, 32 and 64 synchronous parallel processes, and wall-clock time and computational speed-up are reported. MOPLS is used with 4, 8 and 16 synchronous parallel processes with application to the Cannonsville model and results are compared against other efficient MO algorithms including GOMORS, ParEGO and AMALGAM. Results indicate that our algorithms perform considerably well when parallel efficiency and speed-up are considered. The purpose of this study is to illustrate that these efficient parallel algorithms can be effectively used to calibrate the computationally expensive waterways and lake models developed for Singapore (These models are developed in Delft3D [5]). We are currently applying our algorithms for calibration of lake water quality models developed in Delft3D and are successfully running our optimization experiments on NSCC Singapore. In future we plan to run these algorithms asynchronously to further improve computational speed-up and efficiency when running our optimization experiments on supercomputing clusters.

2. Computationally Expensive Water Model Calibration

Recent advances in water resources modeling shows that these models have become increasingly computationally expensive. For instance, within the watershed and hydrological modeling paradigm, the focus has shifted from lumped models to development of distributed and semi-distributed models. These models are computationally expensive (for example, the SWAT model [8]), with simulation times in the order of minutes or even hours. Hydrodynamic and lake water quality models (for example Delft3D [5]) can be even more expensive with simulation times ranging in hours to even days.

Groundwater models are also computationally expensive and typically entails solving complex and computationally expensive Partial Differential Equations (for example, MODFLOW [3]).

Calibration of water resources models, thus, can be a computationally expensive task. Calibration essentially means the adjustment of model parameters such that simulated model response is similar to measured historical response. Automatic methods for model calibration have increased in popularity in recent times where optimization algorithms are used to calibrate model parameters. However, automatic model calibration can be extremely expensive and may require many expensive model simulations.

This study is focused on using high performance / parallel computing and efficient parallel optimization algorithms (e.g, [2,4,7,9,10,11] for calibration of water resources models. We test our parallel and efficient optimization algorithms on two watershed models that simulate flow, sediment and phosphorus in a watershed in upstate New York. These models are referred to as Townbrook and Cannonsville (or Cville) [12].

3. Optimization Algorithms

This study tests one single objective (PR-DDS) and one MO algorithm (MOPLS) within a synchronous parallel optimization framework on the computationally expensive water model calibration problems mentioned previously. The purpose here is to demonstrate the effectiveness and efficiency of these algorithm when used on supercomputers for water resources model calibration.

PR-DDS is a synchronous parallel iterative single-objective search algorithm that initiates via Latin Hypercube Sampling of the parameter domain. The initial samples are then evaluated in parallel via expensive water model simulation. The algorithm then enters the iterative loop where multiple points are selected and subsequently evaluated in parallel in each algorithm iteration. PR-DDS is an efficient parallel version of the Dynamically Dimensioned Search algorithm [13].

MOPLS [1] is a synchronous parallel surrogate-assisted iterative search multi objective optimization algorithm. MOPLS also initiates via Latin Hypercube Sampling and subsequent parallel evaluation of initial samples. Within the iterative loop, MOPLS 1) Fits inexpensive surrogate models for each objective given the already sampled points using Radial Basis Functions, 2) uses non-domination ranking and Tabu search to choose P 'center points' (P denotes the number of parallel simulations for the algorithm) from already sampled points and 3) performs surrogate-assisted search in the neighborhood of each 'center point' to choose P points for expensive evaluation in each algorithm iteration.

Both MOPLS and PR-DDS are designed to run many parallel simulations within each algorithm iteration for improving efficiency. In the experiments, we have tested MOPLS with up to 16 parallel simulations and PR-DDS with up to 64 parallel simulations.

4. Results and Discussion

PR-DDS was tested on the Townbrook model with 8, 16, 32 and 64 processors and results were compared against the serial Dynamically Dimensioned Search (DDS) algorithm [13]. DDS is the serial version of PR-DDS. Figure 1 plots the results obtained by PR-DDS with different processors where the x-axis in the figure denotes the number of parallel iterations completed by the subsequent algorithm and the y-axis denotes the best objective function value found after a specific number of iterations. Results in Figure 1 clearly indicate that PR-DDS is more efficient than serial DDS when parallelization is taken into consideration. PR-DDS with 64 processors has the best performance. Here we can see that PR-DDS gets the result obtained by serial DDS in 100 iterations within 11 iterations. This essentially means that PR-DDS with 64 processors has a 100/11 = 9.1 speed-up in comparison to serial DDS.

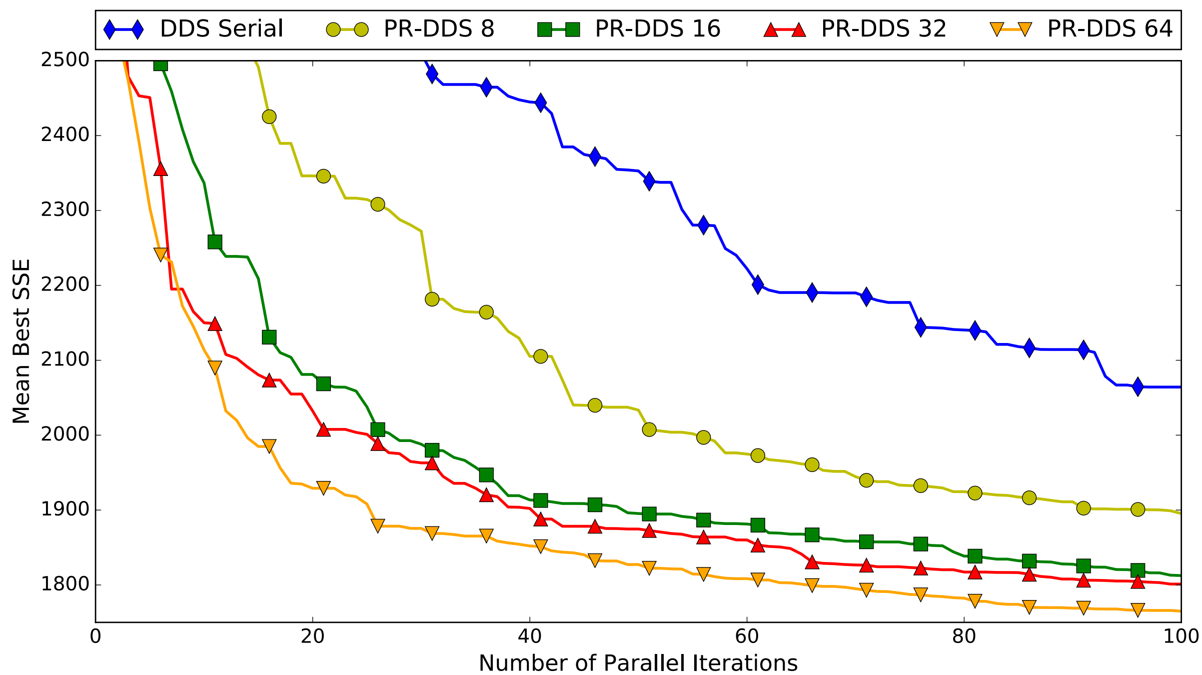

Figure 1: Plots of best objective value found by PR-DDS with different number processors vs number of parallel iterations. (lower curves are better)

The two panels in Figure 2 show similar graphs for two calibration problems based on the Cannonsville case study with application of MOPLS. Here also we are plotting best solution obtained vs number of parallel iterations (used as an approximation of wall clock time). Results of Figure 2 show algorithm results of MOPLS (with 4, 8 and 16 processors), and compare MOPLS against other efficient MO algorithms, namely ParEGO [6], GOMORS [2] and AMALGAM [14]. It is clear from this figure that MOPLS with 16 processors is most efficient. If the serial algorithm ParEGO is considered as baseline, MOPLS-16 gets the same result that ParEGO gets after 60 iterations, after only 5 iterations (approximately) for both optimization calibration problems. This essentially means that MOPLS with 16 processors has a 16-fold speed-up in comparison to ParEGO.

Table 1 Reports the efficiency of MOPLS with 8 and 16 processors when MOPLS with 4 processors is used as baseline. As number of processors increase to 16 MOPLS achieves efficiency greater than 50% in both problems and a super-linear speed-up for one problem (efficiency greater than 100%). These are very promising results and indicate that MOPLS can be used with more parallel processors in future.

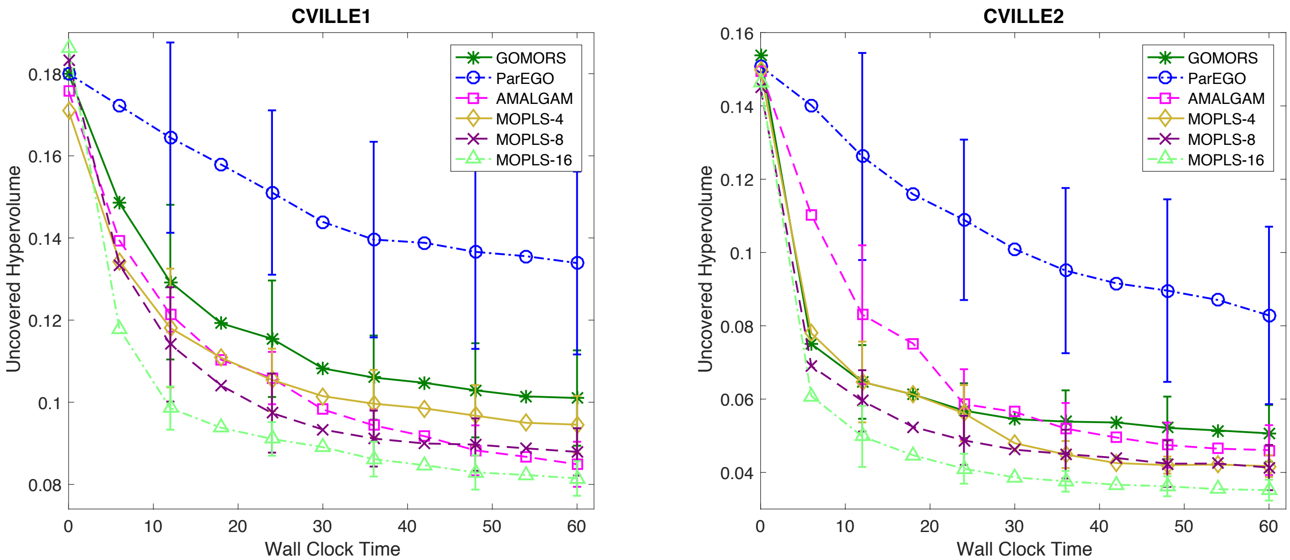

Figure 2: Average uncovered hypervolume versus wall clock time for MOPLS with different number of processors and compared against other algorithms

Table 1: Comparison of efficiency of MOPLS with increase in number of processors

# of processors	**4**	**8**	**16**
CVILLE1 Efficiency	100%	109%	104%
CVILLE2 Efficiency	100%	50%	68%

5. Conclusions and Future Work

Our study illustrates that efficient optimization algorithms and High Performance Computing can be effectively used for optimization and calibration of computationally expensive water resources models. In future we will attempt to calibrate hydrodynamic and lake water quality models developed for Singapore's waterways (typically developed in Delft-3D) using HPC resources of NSCC and by applying our parallel efficient algorithms. We have successfully initiated some of our calibration optimization experiments on Delft3D models using NSCC's computing resources. Many of these models have varying simulation times depending on values of input parameters. Hence, we will incorporate asynchronous parallelization in our future optimizations and model calibration work.

Acknowledgements: This research is funded by US NSF CISE directorate grant to Shoemaker at Cornell University and in Singapore by E2S2 CREATE grant and NUS start-up grant to Prof. Shoemaker.

References:

[1] T. Akhtar (2015), “Efficient Multi-Objective Surrogate Optimization of Computationally Expensive Models with Application to Watershed Model Calibration” (Doctoral Thesis). Cornell University, Ithaca, NY.

[2] Akhtar, T., and C. Shoemaker (2016), “Multi objective optimization of computationally expensive multi-modal functions with rbf surrogates and multi-rule selection”, Journal of Global Optimization, 64(1), 17–32

[3] Arlen W. Harbaugh, MODFLOW-2005, The U.S. Geological Survey Modular Ground-Water Model—the Ground-Water Flow Process

[4] Eriksson, D., D. Bindel, C.A. Shoemaker pySOT (see GITHUB) 2015 (over 16,000 downloads)

[5] Deltares, “Delft3D”, https://oss.deltares.nl/web/delft3d/about, 2011

[6] Knowles, J. (2006), “ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective opti- mization problems”, IEEE Transactions on Evolutionary Computation, 8(5), 1341–66

[7] Krityakierne, T., Akhtar, T. & Shoemaker, C.A. (2016), “SOP: parallel surrogate global optimization with Pareto center selection for computationally expensive single objective problems”, Journal of Global Optimization 66(3), 417-437

[8] Neitsch, S., J. Arnold, J. Kiniry, and J. Williams (2005), “Soil and Water Assessment Tool theoretical documentation version 2005”, US Department of Agriculture Agricultural Research Service.

[9] Regis, R and C. A. Shoemaker. “A Stochastic Radial Basis Function Method for the Global Optimization of Expensive Functions”, INFORMS Journal on Computing, Vol. 19, No. 4, pp. 497-509, 2007

[10] Regis, R.G., C.A. Shoemaker, “Parallel Stochastic Global Optimization Using Radial Basis Functions,” INFORMS Jn. Of Computing 21(3), 411-426, 2009

[11] Regis, R, C.A. Shoemaker, “Combining Radial Basis Function Surrogates Dynamic Coordinate Search in High Dimensional Expensive Black-box Optimization”, Engineering Optimization 45 (5) 529-, 2013

[12] Tolson, B., and C. Shoemaker (2007a), “Cannonsville reservoir watershed swat2000 model development, calibration and validation”, J. of Hydrology, 337, 68–86

[13] Tolson, B., and C. A. Shoemaker (2007b), “Dynamically dimensioned search algorithm for computationally efficient watershed model calibration”, Water Resources, 43, 1–16

[14] Vrugt, J. a., and B. a. Robinson (2007), “Improved evolutionary optimization from genetically adaptive multimethod search”, Proceedings of the National Academy of Sciences of the United States of America, 104 (3), 708–11