Autocalibration of Environmental Process Models using a PAC Learning Hypothesis

Markiyan Sloboda, David Swayne

University of Guelph, Guelph N1G2W1, Canada
dswayne@uoguelph.ca

Abstract. Using the probably approximately correct (PAC) learning hypothesis, we have conducted experiments using clustered computers, high-performance workstations and ad-hoc grids of personal computers, to develop an analytical model for, and demonstrate asymptotic convergence of simple parallel search in the parameter space of complex environmental models such as the Soil and Water Assessment Tool (SWAT). SWAT calibration for hydrological flow, N and P is, for our test cases, superior to current genetic algorithms, as well as to SWAT-CUP, a multi-paradigm calibration solver and to its components. With more complex models, there is no current alternative to our approach in a realizable wall-clock time.

Keywords: SWAT, autocalibration, high performance computing, distributed computing

1 Introduction

Environmental models are widely used for analyzing and predicting physical systems. Typically, models require many variables to simulate real world scenarios, thus leading to increasingly complex computations and as the result to dramatic increase in running time.

In order for the model to predict correct results, it has to be adjusted to the specific region of interest. The most common way to calibrate environmental models is a manual approach. The process is monotonous, slow and requires a lot of expertise in the modeled region as well as considerable expertise. Recently, some automatic calibration tools have been developed, such as genetic algorithms (GA) and stochastic algorithms. Most of these approaches, however, require a complex initial set up, and they typically run in sequential mode, which leads to long running times.

This paper examines some aspects of autocalibration adapted to high-performance computing (HPC), using machine learning, specifically the notion of Probably Approximately Correct (PAC) learning [4]. Our HPC resource is available from the Shared Hierarchical Academic Research Computing NETwork, located in Ontario, Canada. Our experiments with the Soil and Water Assessment Tool (SWAT) [6] for rainfall / runoff estimation in watersheds, and with complex interconnected

hydrological and pollutant transport models such as PolTra and OneLay [5] have led us to the conclusion that an embarrassingly parallel search strategy is an effective way to harness the power of HPC in fitting models to existing observations. Furthermore, a naïve multiobjective fitting strategy for the combination of runoff and water chemistry in SWAT gives acceptable results so long as all of the components (runoff and chemistry) are fitted simulataneously.

2 "Goodness Of Fit" Measure For Hydrological Models

We use both the Coefficient of Determination (CoD or r^2) and the Nash-Sutcliffe Coefficient of Efficiency (NSE) given respectively by:

$$r^2 = \frac{\sum_{i=1}^{N}(O_i - \overline{O})(P_i - \overline{P})}{\sqrt{\sum_{i=1}^{N}(O_i - \overline{O})^2}\sqrt{\sum_{i=1}^{N}(P_i - \overline{P})^2}}, \quad (1)$$

and

$$NSE = 1.0 - \frac{\sum_i (O_i - P_i)^2}{\sum_i (O_i - \overline{O})^2}, \quad (2)$$

where O_i and P_i are observed and model simulated data at time stamp i respectively and the overbar denotes the observed mean of the entire time period of the evaluation. The CoD ranges from 0 to 1 and NSE ranges from minus infinity to 1 (from poor to perfect). NSE represents an improvement over CoD since it is responsive to differences in observed and model-simulated means and variances [3].

3 The Method

Russell and Norvig [4] state that the main PAC principle is based on the following: "any hypothesis that is seriously wrong will almost certainly be "found out" with high probability after a small number of examples, because it will make an incorrect prediction. Thus, any hypothesis that is consistent with a sufficiently large set of training examples is unlikely to be seriously wrong: that is, it must be probably approximately correct".

The main question answered by a PAC-learning algorithm is the determination of the minimum number of examples required.

$$N \geq \frac{1}{\varepsilon}(\ln(H) - \ln(\delta)) \quad (3)$$

where H is the space of all possible hypotheses, and if a learning algorithm returns a hypothesis that is consistent with N examples, then with probability at least 1-δ, it has error at most ε. We typically run the PAC learning approach with δ and ε set at 5% or 1%..

With this in mind, we grid the set of tuneable parameters, and select from the set of all possible configurations a number of parameter sets at least as large as the estimate of N in equation 3. We run those simulations and store in a database for possible future use, and choose on the basis of equation 1 or 2 the best candidate or set of candidates. The process is embarrassingly parallel and, with the aid of a high performance workstation or a cluster of computers it is faster and (in our experience) more accurate than other methods we have tried, including Shuffled Complex Evolution, and the calibration tools known as GLUE [2] and SUFI-2 [1].which form a major part of SWAT-CUP [7].

4 Data Used in Experiments

For calibrating SWAT model data for Raisin River watershed in Southeastern Ontario, Canada was used. Data was available from three Water Survey of Canada (WGC) stream gauging (hydrometric) stations, which are located within the Raisin watershed. Flow values from the station 02MC001 nearest to the outlet from the watershed was used.

Observed data was provided by the Ontario Ministry of Environment, Raisin River Conservation Authority and Water Survey of Canada for research purposes only. Monthly averages for 1985-1994 were used for calibration and for 1995-2004 for validation

5 SWAT Manual Calibration

SWAT model actuators that were varied included: SFTMP, SMTMP, SMFMX, SMFMN, TIMP, SNOCOVMX, ESCO and SURLAG. We used monthly average values for calibration and validation. The manual calibration was conducted by colleagues at Environment Canada. The r^2 and NSE for the monthly calibration were 0.86 and 0.84, respectively. These values are greater than 0.5 and confirm reasonable model results. Validation for TN and TP produced slightly lower, but acceptable, values for NSE.

6 SWAT-CUP

SWAT-CUP is a freely available computer program which calibrates the SWAT model by linking it to several calibration algorithms, such as the Generalized Likelihood Uncertainty Estimation (GLUE), Sequential Uncertainty Fitting (SUFI-2) among others. It provides a user friendly interface for sensitivity analysis, calibration,

validation, and uncertainty analysis of SWAT using only one approach at a time. The following results were obtained for GLUE and SUFI-2.

Table 1: GLUE calibration results for flow, TP and TN Loads.

r^2 Flow	NSE Flow	r^2 TP Load	NSE TP Load	r^2 TN Load	NSE TN Load
0.87	0.85	0.66	-0.06	0.79	0.67

Table 2: GLUE validation results for flow, TP and TN Loads.

r^2 Flow	NSE Flow	r^2 TP Load	NSE TP Load	r^2 TN Load	NSE TN Load
0.81	0.81	0.61	-0.13	0.69	0.59

Table 3: SUFI-2 calibration results for flow, TP and TN loads.

r^2 Flow	NSE Flow	r^2 TP Load	NSE TP Load	r^2 TN Load	NSE TN Load
0.87	0.85	0.55	0.29	0.79	0.67

Table 4: SUFI-2 validation results for flow, TP and TN loads.

r^2 Flow	NSE Flow	r^2 TP Load	NSE TP Load	r^2 TN Load	NSE TN Load
0.85	0.85	0.47	-0.05	0.76	0.67

Flow results and TN load are acceptable and consistent for both calibration and validation and results are very close to manual or GLUE calibrated results. TP load values for calibration time period can be acceptable, since r^2 is greater than 0.5 but NSE = 0.29. However, validation results for TP load are poor, since NSE is negative, generally considered unacceptable.

7 SWAT PAC Learning

In this Section, we show that a so-called *gridded calibration*, at least for this watershed, is capable of producing an equivalent answer.

The first step in gridded calibration was to create a grid of actuators, define range and step values for them. Next, for each separate calculation of the SWAT model, actuators are randomly selected from the grid. Actuators are independent of each other and their values are selected only from the grid points using the uniform distribution. After a certain number of simulations dictated by the PAC learning hypothesis, calibration sensors are sorted in increasing value of the objective function defined, and the set of actuators which satisfy the objective function the best are calibrated values.

Since, calibration evaluations for flow, TP and TN loads were calculated at the same time, it was necessary to use a rule to know if the database contains the best result, and if no other improvements can be made to it. Therefore, a simple weighting process was used. All comparison was done based on the NSE values. The weighted

NSE adopted was 50% flow, 25% TN and 25% TP. Lower values for the flow CoD and NSE are obtained, but all three measures are acceptable, particularly the NSE for TN and TP. Results displayed in the Tables 5 and 6 show high NSE values for flow, TP and TN loads. All these values are above 0.5 and therefore the calibration is successful.

Table 5: Calibration of flow, TP and TN loads, 1% error (3223 simulations)

r^2 Flow	NSE Flow	r^2 TP Load	NSE TP Load	r^2 TN Load	NSE TN Load
0.86	0.82	0.68	0.62	0.75	0.69

Table 6: Validation results for flow, TP and TN loads with 1% error

r^2 Flow	NSE Flow	r^2 TP Load	NSE TP Load	r^2 TN Load	NSE TN Load
0.87	0.85	0.58	0.27	0.69	0.61

To run each SWAT simulation without any modifications using Raisin River watershed dataset, it requires about 4.5 minutes on the average desktop computer and about 30 seconds on most clusters on the SHARCNET, with our parallelized (MPI) SWAT version. Even, with such a short run time it would take a long time to generate the 10^{12} possible values. Therefore, the space was scaled down to 2 million distinct actuator sets. The simulations corresponding to this space were generated, when processors on the SHARCNET clusters were available. It took a week, using 200 processors.

Since, the new space $\overline{H}$ is the sub-space of *H*, there is a need to calculate a *shift* in the number of required simulations, which we derived as:

$$N - \overline{N} \geq \frac{1}{\varepsilon}\left(\frac{10^{12}}{2 * 10^{6}}\right) = \frac{1}{\varepsilon} * 13.12 \qquad (4)$$

where N – total number of simulations required, when the whole space $H = 10^{12}$ is used

$\overline{N}$ – total number of simulations, when the sub space $\overline{H} = 2 * 10^{6}$ is used

ε – the upper error value

For the original 10^{12} simulations, 5% error would require 613 iterations and 1% error would require 3223 simulations. The minimum number of simulations required to satisfy the PAC learning theorem for the subspace space $\overline{H}$ is, for 1% error at least 1912 calls to the SWAT model and for 5% error at least 350 simulations.

Since, calibration evaluations for flow, TP and TN loads were calculated at the same time, a simple weighting process was used. All comparison was done based on the NSE values, since the NSE represents the best measure of the closeness of the simulated results to the observed data. The weighted NSE adopted was 50% flow,

25% TN and 25% TP. The highest $Weighted\ Total\ NSE$ from all three components within the 2 million records in the database was equal to 0.77.

To analyze how results are correlated to the number of simulations and to confirm that a PAC learning theorem is acceptable it was decided to run different number of simulations 1000 times each. The number of simulations being tested were from 500 to 5000 with the 500 iterations interval, i.e. 500, 1000, 1500,…, 4500, 5000. The highest $Weighted\ Total\ NSE$ from all three components within the 2 million records in the database was equal to 0.77. We approximated this experiment by building a database containing 2 million records, we did not re-run SWAT simulations, but instead actuators and corresponding results were randomly chosen from the database of 2 million simulations, using the best NSE for flow only.

The best flow-only result over all 2 million simulations (NSE = .986011) is already stored in the database. This value was taken as a benchmark to which all the other results were compared. Q1 is the lower quartile (25^{th} percentile), Q3 is the upper quartile (75^{th} percentile) and IQRange is the interquartile range.

Table 7: Box plot statistics of NSE for percent accuracy (flow only).

Sample size	Q1	Median	Q3	IQRange	Whiskers (from, to)
315 (95%)	0.866952	0.883709	0.905719	0.0387668	(0.809034, 0.961108)
500	0.873536	0.891648	0.914876	0.0413396	(0.823719, 0.963922)
1912 (99%)	0.908964	0.924717	0.938933	0.0299687	(0.865485, 0.982025)
4000	0.926839	0.938907	0.951072	0.0242329	(0.891142, 0.986011)
4500	0.928043	0.941754	0.954204	0.0261607	(0.891745, 0.986011)
5000	0.930305	0.942234	0.953604	0.0232990	(0.897061, 0.986011)

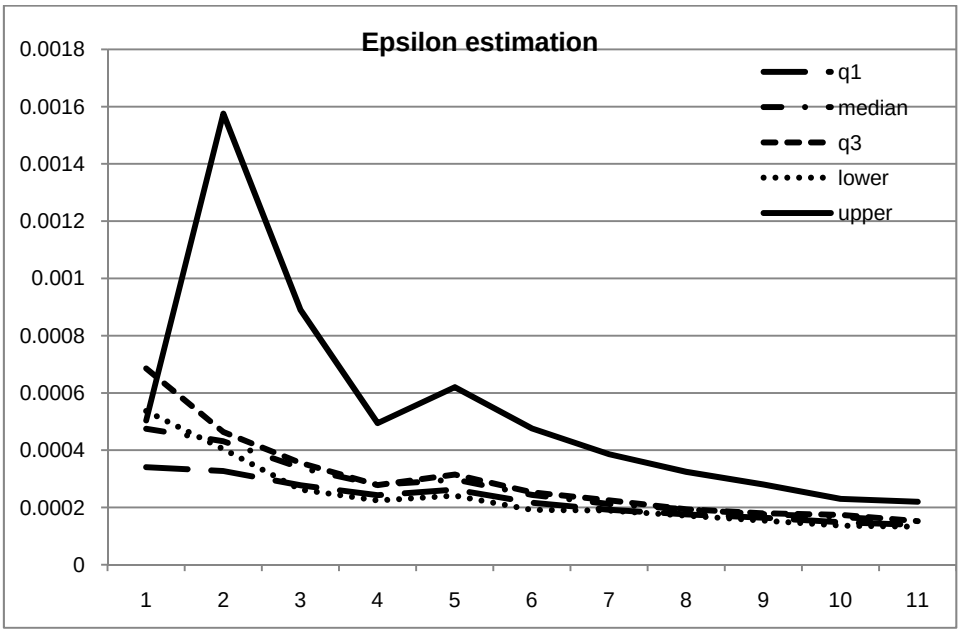
The (IQR) interquartile narrows as the sample sizes increase, and the full range of NSE values at and above 99% is significantly higher than from the other methods as calculated

In a further analysis, we derived a tighter approximation to the actual ε and δ in the PAC theorem and we developed a relationship between the δ^* for each of the calculated N^* values in Table 7, as a ratio of the baseline δ value for $N = 315$:

$$\delta/_{\delta^*} = e^{\varepsilon(N^*-N)}, \text{ and } \varepsilon = \frac{\ln((1-NSE)/(1-NSE^*))}{N^*-N}.$$

Experimentally, we find that, as N increases, the value of ε stabilizes (*Figure 1*).

Figure. 1. Figure 1: Epsilon estimation



8 Conclusions

Table 8: Comparative run-time for all tested techniques.

Technique	Computation	Run time
GLUE	Server	15 days, 11 hours, 35 min
SUFI-2	Server	3days 2 hours, 30 min
Explicit gridded with 1% error	Laptop	4 days 20 hours 26 min
Explicit gridded with 5% error	Laptop	22 hours 11 min
Explicit gridded with 1% error	Server	8 hours 9 min
Explicit gridded with 5% error	Server	1 hour 32 min
Explicit gridded with 1% error	SHARCNET	1 hour 6 min
Explicit gridded with 5% error	SHARCNET	10 min 24 sec

This paper has examined some aspects of autocalibration adapted to high-performance computing (HPC), using machine learning, specifically the notion of Probably Approximately Correct (PAC) learning. Our experiments with the Soil and Water Assessment Tool (SWAT) for rainfall / runoff estimation in watersheds, and with complex interconnected hydrological and pollutant transport models have led us to the conclusion that an embarrassingly parallel search strategy is an effective way to harness the power of HPC in fitting models to existing observations. Furthermore, a

naïve multiobjective fitting strategy for the combination of runoff and water chemistry in SWAT gives acceptable results so long as all of the components (runoff and chemistry) are fitted simultaneously. Calibration times are reduced from days or weeks, to hours, depending on the availability of high performance computing resource (multi-core server or computer cluster such as SHARCNET), as shown in Table 8.

REFERENCES

1. Abbaspour, K.C., Johnson, A., van Genuchten, M.T. 2004. Estimation of uncertain flow and transport parameters by a sequential uncertainty fitting procedure: SUFI-2. Vadose Zone Journal 3 (4). pp. 1340–1352.
2. Beven, K., Freer, J. 2001. Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems using the GLUE methodology. Journal of Hydrology 249. pp. 11–29.
3. Legates, D., McCabe, J. 1999. Evaluating the use of Goodness of Fit measures in hydrological and hydroclimatic model validation. Water Resources Research 35. pp. 233-241.
4. Russell, S., Norvig, P. 2003. Artificial Intelligence: a Modern Approach, 2nd Ed., by Stuart Russell and Peter Norvig, Prentice-Hall, USA.
5. Simons, T.J., Lam, D.C.L. 1986. Documentation of a Two-Dimensional X-Y Model Package for Computing Lake Circulations and Pollutant Transports. American Society of Civil Engineers, New York, NY, pp. 258-267.
6. SWAT - www.brc.tamus.edu/swat.
7. SWAT-CUP - www.eawag.ch.