

# Agile tuning method in successive steps for a river flow simulator

Mariano Trigila<sup>1,2</sup>, Adriana Gaudiani<sup>2,3</sup>, and Emilio Luque<sup>4</sup>

<sup>1</sup> Facultad de Informática, Universidad Nacional de La Plata, Buenos Aires, Argentina  
marianotrigila@gmail.com

<sup>2</sup> Facultad de Ingeniería y Ciencias Agrarias, Pontificia Universidad Católica Argentina, Ciudad Autónoma de Buenos Aires, Argentina  
mariano\_trigila@uca.edu.ar

<sup>3</sup> Instituto de Ciencias, Universidad Nacional de General Sarmiento, Buenos Aires, Argentina  
agaudi@ungs.edu.ar

<sup>4</sup> Depto. de Arquitectura de Computadores y Sistemas Operativos, Universidad Autónoma de Barcelona, 08193 Bellaterra (Barcelona) España  
emilio.Luque@uab.es

**Abstract.** Scientists and engineers continuously build models to interpret axiomatic theories or explain the reality of the universe of interest to reduce the gap between formal theory and observation in practice. We focus our work on dealing with the uncertainty of the input data of the model to improve the quality of the simulation. To reduce this error, scientist and engineering implement techniques for model tuning and they look for ways to reduce their high computational cost. This article proposes a methodology for adjusting a simulator of a complex dynamic system that models the wave translation along rivers channels, with emphasis on the reduction of computation resources. We propose a simulator calibration by using a methodology based on successive adjustment steps of the model. We based our process in a parametric simulation. The input scenarios used to run the simulator at every step were obtained in an agile way, achieving a model improvement up to 50% in the reduction of the simulated data error. These results encouraged us to extend the adjustment process over a larger domain region.

**Keywords:** Parametric simulation, tuning methodology, flood simulation improvement, dynamical systems, flood model calibration.

## 1 Introduction

The models built by scientists and engineers are often expressed in terms of processes that govern the dynamics and in terms of entities that determine the system states. They usually implement the models in a computerized simulation system. The values provided by a simulation are the model response to a certain system scenario [8]. Tuning, calibration or adjustment of the simulator are improvement process that seek the best set of input parameters to achieve the smallest difference between the output

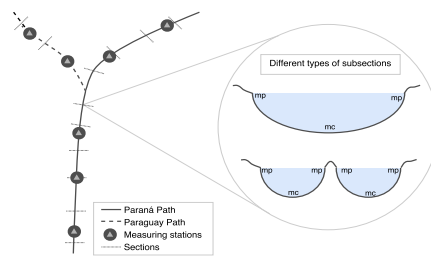
data and the reference data set [1][6]. The automatic tuning of a simulation needs multiple instances of simulator running, one for each parameters combination. In consequence, the more parameters are considered, the more computing time is needed [8]. The main idea proposed in this article is an automatic tuning methodology for a simulator of a complex dynamic system that models the waves' displacement in rivers and channels. This approach exploits a local behavior of the system: Parameters of the domain of the system with spatial proximity do not change or differ very little, which allows us to reduce the search space and in consequence, the computational cost. We take advantage of the research and the results of previous works [2][3]. Using our methodology, we were able to find input scenarios to execute the simulation that provided an improvement of up to 50% in the quality of the simulation in relation to the initial scenario (currently used for simulation and forecasting).

## 2 Description of the simulator

The computational model used in our experiments is a simulation software developed in the Laboratory of Computational Hydraulics of the National Institute of Water (INA). This computer model calculates the translation of waves through a riverbed calculated by the equations of Saint Venant. It implements a one-dimensional hydrodynamic model of the Paraná River for hydrologic forecasting [4][5]. Next, we describe the key features of the computational model.

### 2.1 Domain modeling feature

Simulator represents a hydrodynamic model consisting of two sections or filaments. Each filament represents the path of a river. See graphical representation in Fig. 1. To simulate the transport of water in a filament channel, its route is subdivided into sections. Each section ( $Sc$ ) represents a specific position within the path, and it is divided into subsections ( $Su$ ).



**Fig. 1.** Discretization of the river domain. Some types of  $Su$  in the cross  $Sc$  of the domain.

The simulator requires setting a set of input parameters values, which determines a simulation scenario. At every  $Su$ , the roughness coefficient of Manning ( $m$ ) is the parameter used as an adjustment variable, which depends on the resistance offered by the main channel and the floodplain [1][5]. We distinguish both values as Manning of plain ( $mp$ ) and Manning of channel ( $mc$ ). Depending on the channel geometry in

each section, a greater or lesser amount is needed of  $mp$  and  $mc$ . The different forms of the sections were shown in Fig. 1.

## 2.2 Observed data measured at monitoring stations

A monitoring or measuring station (St) is the "physical and real" place where the river heights are surveyed and recorded. Each St is located in a city on the banks of the river channel. The data collected and recorded from the height of the river are known as observed data (OD), and they are measured daily. Data for the period from 1994 to 2011 were used to implement the experiences carried out in this work [4].

## 3 Proposed methodology

We propose a search methodology to improve a simulator quality by finding the best set of parameters. Our aim is to optimize the simulation for a reduced search space,  $\omega$ , such that  $\omega \subset \Omega$ , minimizing the use of computing resources to achieve the objective, where  $\Omega$  is the whole search space with all possible combinations of the adjustment parameters and  $\omega$  is the resulting reduced space [7]. Unlike the work in [3], we propose a calibration process of successive tuning steps to obtain an adjusted input parameters values from a preselected set of successive sections. Each parameters combination determines a simulation scenario and we detail its structure below. The quality of the simulated data ( $SD$ ) is measured through calculating a divergence index ( $DI$ ), as we explain in section 3.4.

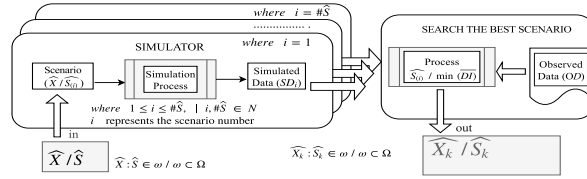


Fig. 2. Search process of the best scenario  $\widehat{S}_k$  for  $k$  station.

We start the method by choosing a monitoring station  $St_k$  located in an initial place  $k$  on the riverbed and selecting three contiguous sections, which are adjacent to that station. Fig. 2 shows an outline of the set of possible scenarios and the selection of the best of them. This process searches the adjusted parameters set,  $\widehat{X}$ , for the station  $k$ , which determines the best simulation scenario  $\widehat{S}_k$ . This tuning method is repeated for the next station in a successive way, extending the successive adjustment process until reaching the last one, as we show in Fig. 3.

### 3.1 Structure of the input scenario

We define a set of parameters for section  $m$ , subdivided into three subsections, as the 3-tupla:

$$Sc_m = (mp_m, mc_m, mp_m) \quad (1)$$

Where  $mp_m$  represent Manning value of plain and  $mc_m$  represent Manning value of channel. We remark that equation (1) has two independent variables,  $mp_m$  and  $mc_m$  and it takes the same  $mp_m$  value at both section ends. Initially, three contiguous and adjacent sections were chosen for station  $k$ , being its scenario  $\widehat{S}_k$  defined by:

$$\widehat{S}_k = \begin{bmatrix} Sc_m \\ Sc_{m+1} \\ Sc_{m+2} \end{bmatrix} = \begin{bmatrix} mp_m & mc_m & mp_m \\ mp_{m+1} & mc_{m+1} & mp_{m+1} \\ mp_{m+2} & mc_{m+2} & mp_{m+2} \end{bmatrix} = \begin{bmatrix} mp_k & mc_k & mp_k \\ mp_k & mc_k & mp_k \\ mp_k & mc_k & mp_k \end{bmatrix} \quad (2)$$

Being a physical system, and because the sections are close together, it is assumed that the three sections have the same values of  $mp$  y  $mc$  for  $St_k$ .

We remark in Equations (4) and (5) that,  $mp_k$  and  $mc_k$  are independent variables. Therefore, the input scenario used to start the tuning process  $\widehat{X}$  is determined by the scenarios  $\widehat{S}_k$  corresponding to the sections  $Sc_m$ , and for the intermediate scenarios  $\widehat{S}_k^+$  corresponding to the intermediate sections  $Sc_m^+$  located between the stations  $k$  y  $k + 1$ . Equation (5) represents  $\widehat{X}$  structure for  $n$  stations:

$$\widehat{X} = \{ \widehat{S}_k, \widehat{S}_k^+, \widehat{S}_{k+1}, \widehat{S}_{k+1}^+, \dots, \widehat{S}_n \}, \text{ with } k = 1 \quad (3)$$

### 3.2 Discretization process parameters.

The variation range of both  $imp$  and  $imc$ , and the increment step,  $smp$  and  $smc$ , are initial set values provided by INA experts. We implemented our experiences based on these values.

$$imp = [mp_{min}, mp_{max}] = [0.1, 0.71]; smp = 0.01 \quad (4)$$

$$imc = [mc_{min}, mc_{max}] = [0.017, 0.078]; smc = 0.001 \quad (5)$$

$$\# \widehat{S} = 61 \mid \frac{mp_{max} - mp_{min}}{smp} = \# \widehat{S} \wedge \frac{mc_{max} - mc_{min}}{smc} = \# \widehat{S} \quad (6)$$

( $\# \widehat{S} = 61$ ) was obtained empirically for us. It represents minimum value of scenarios, which allow us to get improved output values when running the simulation. Increment  $\# \widehat{S}$  could increase the accuracy but will effectively increase the use of computational resources. Equation (9) determines each scenario values  $\widehat{S}_{k(i)}$ :

$$\widehat{S}_{k(i)} = \begin{bmatrix} mp_i & mc_i & mp_i \\ mp_i & mc_i & mp_i \\ mp_i & mc_i & mp_i \end{bmatrix} = \begin{bmatrix} (smp \cdot i) + mp_{ini} & (smc \cdot i) + mc_{ini} & (smp \cdot i) + mp_{ini} \\ (smp \cdot i) + mp_{ini} & (smc \cdot i) + mc_{ini} & (smp \cdot i) + mp_{ini} \\ (smp \cdot i) + mp_{ini} & (smc \cdot i) + mc_{ini} & (smp \cdot i) + mp_{ini} \end{bmatrix} \quad (7)$$

Where  $i$  is the number of scenario, the range  $[i, \#\widehat{S}] \subset \mathbb{N}$ , where  $1 \leq i \leq \#\widehat{S}$ ,  $mp_{ini}$  and  $mc_{ini}$  are the initial values. We use these values to start the search process and to run the simulator for each scenario, in order to find the best one as we describe in next section.

### 3.3 Search of the best scenario

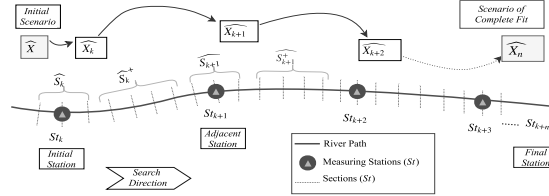
Through a divergence index implemented with the root mean square error estimator (RMSE), the best-input scenario is determined by comparing the SD series with the OD series.

$$DI_k^y = RSME_k^y = \sqrt{\frac{\sum_{i=1}^{i=N} (H_k^{OD,y} - H_k^{SD,y})^2}{N}} \quad (8)$$

The index  $DI_k^y$  is the RMSE error of the series of river heights simulated  $H_k^{SD,y}$  with respect to of the series of river heights observed  $H_k^{OD,y}$ , for a station  $k$ , and for a year  $y$ , which is the simulation time, and the number of stations,  $N$ . The best fit scenario for the  $k$  station which generates a set of output  $H_k^{SD}$  is the minimum  $DI_k^y$ , ( $\min(\overline{DI_k^y})$ ) of all the simulations.

### 3.4 Successive tuning process

After obtaining the scenario that best fits for St  $k$ , the adjustment can be extended to a new St  $k + 1$ . We take advantage of the system locality behavior and we set the parameters with the previously calculated adjustment values. To make this possible, the scenario  $\widehat{S}_k^+$  is initialized with the values of the best scenario  $\widehat{S}_k$ , immediately before. We took advantage of the locality behavior, which means that those sections that are close one to another have similar adjustment parameters values or they differ very little.



**Fig. 3.** Successive tuning process

In the successive input scenarios, we “leave fixed the adjusted parameters values found” in the previous calibrations, and thus the previous adjustment scenarios of each section are used to find the actual adjusted parameters values. For  $k$  St the new  $k$

parameter vector is:  $\widehat{X}_k = \{\widehat{S}_k, \widehat{S}_k^+, \widehat{S}_{k+1}, \widehat{S}_{k+1}^+, \dots, \widehat{S}_n\}$

The  $k + 1$  input scenario, or  $k+1$  parameters vector, is:

$$\widehat{X}_{k+1} = \{\widehat{S}_k, \widehat{S}_k^+, \widehat{S}_{k+1}, \widehat{S}_{k+1}^+, \dots, \widehat{S}_n\}, \text{ where } \widehat{S}_k^+ = \widehat{S}_k$$

The  $k + 2$  input scenario, or  $k+2$  parameters vector is:

$$\widehat{X}_{k+2} = \{\widehat{S}_k, \widehat{S}_k^+, \widehat{S}_{k+1}, \widehat{S}_{k+1}^+, \dots, \widehat{S}_n\}, \text{ where } \widehat{S}_k^+ = \widehat{S}_k, \widehat{S}_{k+1}^+ = \widehat{S}_{k+1}$$

For  $n$  input scenario to the Simulator (scenario that adjusts the entire domain):

$$\widehat{X}_n = \{\widehat{S}_k, \widehat{S}_k^+, \widehat{S}_{k+1}, \widehat{S}_{k+1}^+, \dots, \widehat{S}_n\}, \text{ where } \widehat{S}_k^+ = \widehat{S}_k, \dots, \widehat{S}_{n-1}^+ = \widehat{S}_{n-1} \quad (9)$$

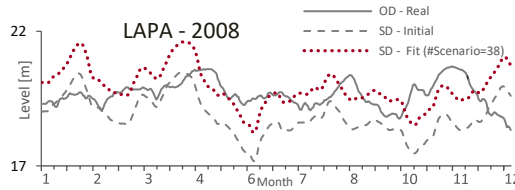
## 4 Experimental Results

In search of the best scenario performed on the  $k$  St “Esquina” (ESQU), we found scenarios that improved the results up to 57% in relation to the initial scenario used by the INA’s experts, determined by ratio of  $DI_k^y(Fit)$  to  $DI_k^y(Initial)$ .

**Table 1.** Adjustment made in  $k$  St and  $k + 1$  St, several years.

$S_i$	$\widehat{S}_{k(\infty)}$	Year	Station	Station ID	$DI_k^y(I) - Initial$	$DI_k^y(F) - Fit$	Improvement $DI_k^y(F)/DI_k^y(I)$
46	(0.55, 0.062, 0.55)	2008	$k$	ESQU	1.55	0.66	57 %
54	(0.63, 0.070, 0.63)	1999	$k$	ESQU	1.42	0.87	39 %
38	(0.47, 0.054, 0.47)	2002	$k$	ESQU	1.24	0.97	22 %
38	(0.47, 0.054, 0.47)	1999	$k+1$	LAPA	1.72	0.94	45 %
30	(0.39, 0.046, 0.39)	2008	$k+1$	LAPA	1.17	0.89	24 %

We show in Table 1 the synthesis process with the top three scenarios found for processed  $k$  St. As also, it shows the second St  $k + 1$  adjusted. The best scenario was found at “La Paz”, St (LAPA), which is an adjacent St to ESQU.



**Fig. 4.** Comparative OD, SD, Fit. Station  $k = LAPA$ ,  $y = 2008$

We show in Table 1, a synthesis process with the two best scenarios found for  $k + 1$  station. Fig. 4 shows a comparative graph with the observed data series (real measured), the initial simulated data series (original series loaded in the simulator) and the series of simulated data adjusted for the best fit scenarios in ( $k + 1$ ) station. We can see that our method achieves global better results over the whole series.

## 5 Conclusions

Our method “in successive steps of adjustments” was tuned by the locality simulation behavior, which provided promising results when finding scenarios that improved simulation quality and these encouraged us to continue our research in this direction. These scenarios provided numerical series of river heights closest to those observed at the measurement stations on the riverbed. The improvement percentage obtained was greater than 50%. The method is simple and manages to reduce the computational resources based on proportional successive increases of the initial scenario parameters (7). Thus, we reduced the time computing when we use the adjusted parameters obtained in the previous steps to calculate the actual one (9). We continue our work focused on finding an automatic calibration methodology of the computational model extending this methodology from a predetermined initial station to the last one at the end of the riverbed and applying the methodology described in this work. This proposal will make use of HPC techniques to decrease the execution times.

### Acknowledgments

The MICINN/MINECO Spain under contracts TIN2014-53172-P and TIN2017-84875-P has supported this research. We are very grateful for the data provided by INA and we appreciate the guidance received from researchers at INA Hydraulic Laboratory.

### References

1. Bladé, E., Gómez-Valentín, M., Dolz, J., Aragón-Hernández, J., Corestein, G. and Sánchez-Juny, M., Integration of 1D and 2D finite volume schemes for computations of water flow in natural channels, *Advances in Water Resources*, Vol. 42, págs. 17-29 (2012).
2. Cabrera, E., Luque, E., Taboada, M., Epelde, F. and Iglesias, M., Optimization of emergency departments by agent-based modeling and simulation. 2012. *Information Reuse and Integration (IRI)*, IEEE 13th International Conference on. págs. 423-430 (2012).
3. Gaudiani, A., Luque, E., García, P., Re, M., Naiouf, M., De Giusti, A., How a Computational Method Can Help to Improve the Quality of River Flood Prediction by Simulation. *Advances and New Trends in Environmental and Energy Informatics*. Switzerland: Springer International Publishing, págs. 337-351 (2016).
4. Menéndez, A., Three decades of development and application of numerical simulation tools at INA Hydraulic lab., *Mecánica Computacional*, Vol. XXI, págs. 2247-2266 (2002).
5. Krauß, T and Cullmann, J., Identification of hydrological model parameters for flood forecasting using data depth measures., *Hydrology and Earth Systems*, Vol. 8, págs. 2423–2476 (2011).
6. Sargent, R., Verification and validation of simulation models. *Journal of Simulation*, Vol. 7, págs. 12-24, (2013).
7. Wang, Long-Fei and Shi, Le-Yuan, Simulation Optimization: A Review on Theory and Applications. 11, *Acta Automatica Sinica*, Vol. 39, págs. 1957-1968 (2013).
8. Wu, Q., Liu, S., Cai, Y., Li, X., Jiang, Y. Improvement of hydrological model calibration by selecting multiple parameter ranges, *Hydrology and Earth System Sciences*, Vol. 21, págs. 393-407 (2017).