Thermo-hydro-mechanical model parameters adjustment via a distributed scientific workflow system

Aldemon Lage-Bonifácio, Rafaela de Oliveira-Amaral, Michèle Cristina Resende-Farage, Ciro de Barros-Barbosa & Flávio de Souza-Barbosa

Programa de Pós-Graduação em Modelagem Computacional, Universidade Federal de Juiz de Fora, Brazil. aldemon.bonifacio@engenharia.ufjf.br, rafaela.amaral@engenharia.ufjf.br, michele.farage@ufjf.edu.br, ciro.barbosa@ice.ufjf.br, flavio.barbosa@engenharia.ufjf.br

Abstract
This work presents a numerical approach for the automatic adjustment of parameters of the Mazars Damage Model, applied to a thermo-hydro-mechanical modeling of concrete structures. The procedure is based on a Scientific Workflow System (SWS) that addresses the combinatorial universe of adjustable parameters by minimizing the number of simulations required for optimized results. Not only does SWS improve efficiency, by also makes the strategy easier when compared to manual procedures. The adopted algorithm is developed in an intuitive script language and employs a distributed computational environment. Comparison to experimental data indicates that the proposed methodology was efficient and effective in improving the analysis, by minimizing errors and saving processing time.

Keywords: Scientific Workflow System, Damage Modeling, Parameters Adjustment.

Modelo termo-hidro-mecánico de ajuste de parámetros mediante un sistema científico de flujo de trabajo distribuido

Resumen
Este trabajo presenta una estrategia numérica para el ajuste automático de parámetros del Modelo de Daño de Mazars, aplicado a un modelado termo-hidro-mecánico de estructuras de concreto. El procedimiento se basa en un sistema de flujo de trabajo científico (SWS) que aborda el universo combinatorio de parámetros ajustables al minimizar el número de simulaciones requeridas para obtener resultados optimizados. SWS no sólo mejora la eficiencia, sino que también facilita la estrategia en comparación con los procedimientos manuales. El algoritmo adoptado se desarrolla en un lenguaje de escritura intuitivo y emplea un ambiente computacional distribuido. La comparación con los datos experimentales indica que la metodología propuesta fue eficiente y efectiva para mejorar el análisis, minimizando los errores y ahorrando tiempo de procesamiento.

Palabras clave: Sistema de flujo de trabajo científico, Modelo de Daño, Ajuste de Parámetros.

1. Introduction

Computational modeling is an important and useful tool for the scientific knowledge applied to several areas of human activity, such as engineering and biological sciences. Its contribution regards economical and ethical aspects of conducting experiments, as well as efficiency improvements related to data manipulation [1].

However, the application of computational methods to simulation, in general, involves theoretical complexity. In general, in-silico experimentation requires researchers to have a multidisciplinary profile - demanding knowledge beyond a specific study area - often related to mathematical and computational techniques.

The present work consists of a numerical procedure for adjusting model parameters. The proposed methodology aims to treat the addressed problem efficiently while reducing the difficulties mentioned above. The approach consists of implementing an operational strategy to minimize the combinatorial space of adjustable parameters and, consequently,

increasing the operational efficiency by means of a Scientific Workflow System (SWS) which automates the process and accelerates its execution via distributed processing [2,3].

In order to address the difficulty inherent to processing on a complex operating environment, it was employed herein a SWS named Scientific Workflow Automation in Distributed Environment (SWADE). SWADE is a system based on web services which allows the researcher to focus attention on the development and execution of the workflow, minimizing the handling of distributed and heterogeneous computing environment technical concepts.

SWADE was applied herein to the estimation of a number of characteristic parameters of the Mazars Mechanical Damage Model [4] via inverse analysis procedures. Those parameters were obtained from experimental curves obtained from concrete specimens subjected to high temperatures [5].

The following sections present: an overview of important concepts and definitions dealt in the present work; an architectural view of the *ad hoc* SWS; the application of SWADE to a Computational Mechanics problem and, finally, some conclusions on the results obtained with the adopted approach.

2. Important concepts and definitions

2.1. Overview of the experimental data

The experimental data employed as reference herein were obtained at the *Laboratoire de Mécanique et Matériaux du Génie Civil* (L2MGC), University of Cergy-Pontoise, France [6,7]. Tested specimens consisted of concrete-rock bilayers made of conventional concrete and high performance concrete and limestone (Fig. 1), aiming to reproduce a typical sample of a tunnel structure. The goal was to submit the double layered blocks to temperatures up to 600 °C and 750 °C, measuring the temperature evolution at specific points along the height and observing the different effects of temperature on the two types of concrete, rock and interface region.

Fig. 2 shows the scheme adopted to monitor the temperature evolution during the thermal tests. For each sample five thermocouples were installed along the central axis of the prismatic sample, arranged in the following positions: the upper surface of the concrete, half of concrete layer, rock-concrete interface, half of rock layer and the bottom surface of the rock. The blocks were isolated with fiberglass (except for the upper surface, which was directly exposed to heating) so as to assure one-directional heating - for simplification purposes concerning the numerical simulations. Fig. 3, 4 show the typical aspect of the concrete blocks after heating, clearing denoting the damages.

It well known that temperature evolution depends on the integrity of the tested materials. Hence, experimental information obtained from these tests were adopted in the present study, as described in section 2.2.

Further details concerning the experimental procedures and the THM model are available in references [7-9].

2.2. Parameters Adjustment for Mazars Model

Concrete is a material that suffers mechanical degradation when exposed to high temperatures. This effect may be represented by means of damage models, such as the Mazars Model [4]. In outline, the model assumes that tensile

deformations lead to distributed microcracks in the material, which causes degradation and affects the macrostructural behavior of the concrete.

The Mazars Damage is represented by a scalar quantity named Damage variable, *D*, dependent on six characteristic parameters obtained from destructive mechanical tests accomplished on concrete samples: A_t, B_t, A_c, B_c, ε_{d0} and β. These six parameters are extracted for stress-strain curves. The *D* variable is given by eq. (1)-(3).
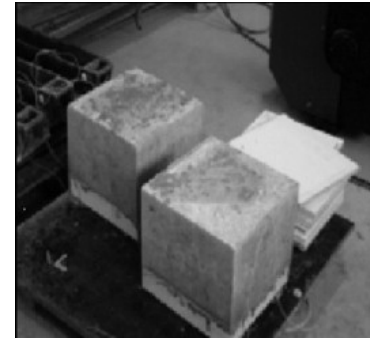

Figure 1. Bilayer samples made of concrete and rock.
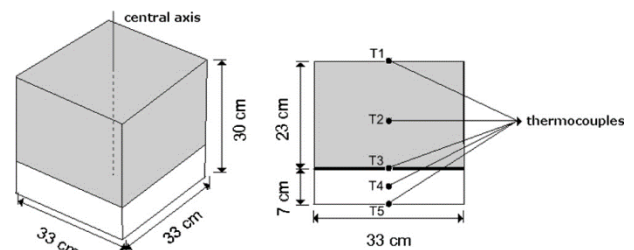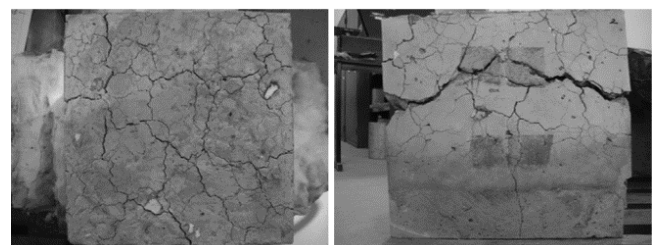Source: Ferreira, 2014.

Figure 2. Geometry of samples and location of thermocouples.
Source: Ferreira, 2014.

(a) Heated surface   (b) Lateral surface
Figure 3. CC-600 after heating.
Source: Ferreira, 2014.

(a) Heated surface   (b) Lateral surface
Figure 4. CC-750 after heating.
Source: Ferreira, 2014.

$$D_t = \left(1 - \frac{\varepsilon_{d0}(1 - A_t)}{\hat{\varepsilon}} - \frac{A_t}{\exp[B_t(\hat{\varepsilon} - \varepsilon_{d0})]}\right) \quad (1)$$

$$D_c = \left(1 - \frac{\varepsilon_{d0}(1 - A_c)}{\hat{\varepsilon}} - \frac{A_c}{\exp[B_c(\hat{\varepsilon} - \varepsilon_{d0})]}\right) \quad (2)$$

$$D = \alpha_t^\beta D_t + \alpha_c^\beta D_c \quad (3)$$

where $\hat{\varepsilon}$ is equivalent deformation, β is a scalar value intrinsic to the material, and $\alpha_t^\beta + \alpha_c^\beta = 1$. D assumes values ranging from 0 to 1 and is a means of quantifying the mechanical damage through gradual decreasing the elasticity module. Eq. (4) indicates the adapted Hooke Law for such a condition.

$$\sigma = E_0(1 - D)\epsilon \quad (4)$$

where σ is the stress, ε is the strain and E_0 is the elasticity module of the undamaged material.

In spite of the simplicity of the model expressed by eq. (1)-(4), the experimental parameters A_t, B_t, A_c, B_c, ε_{d0} and β are not easy to measure. It is due to the brittle characteristic of concretes, which makes it rather difficult to obtain the appropriate stress-strain curve under tension. An alternative methodology consists of inverse analysis - which is possible with the help of experimental data concerning E evolution under varied deleterious solicitations.

The present experiment consisted of the evaluation of the six Mazars parameters from experimental data comprising the evolution of E for a concrete subjected to varying temperatures, obtained by [6]. Then, experimental evolution of the damage variable D is given by eq. (5).

$$D(T) = 1 - \frac{E(T)}{E_0} \quad (5)$$

The mechanical behavior of the concrete structures under high temperatures was simulated through a transient nonlinear thermo-hydro-mechanical model (THM) and the Mazars Model. The software applied to the analysis was the CAST3M, which is an open source Finite Element code developed in the programming language Gibiane [10]. Since experimental information was not available to directly identify the Mazars parameters, it was adopted an adjustment procedure consisting, basically, on the following steps:

- applying the THM model, via CAST3M, with a set of Mazars parameters;
- numerical evaluation of temperature evolution curves;
- comparison of the numerical curves to their experimental counterparts.

2.3. Scientific Workflow

A scientific workflow is a description of the steps needed to execute a particular experiment, including its artifacts and processes. A scientific workflow system (SWS) is a computational tool to automate workflows' execution, helping researches to create and execute computational experiments, giving access to services and information available in a computing environment [11]. However, other definitions may be found in [12-14]. The key feature of SWSs is the flexibility, i.e., making it easy to maintain workflows; employing heterogeneous data; tracking execution; and analyzing results. These characteristics are paramount to the efficiency of research activities [15].

A large number of SWS tools are available for use [16], but these tools are focused on specific set of functionalities. As an example, VisTrails is oriented towards an efficient mechanism for tracking experiments, also known as provenance [17]. Kepler presents significant results on the interface to handle building blocks of experiments, as well as the composition of computational models [18]. However, the methodology proposed herein presents important characteristics such as the possibility to extend the set of executable actions; abstraction of the technological complexity; and a distributed and heterogeneous computational environment.

3- SWADE Architecture

The lightweight code of SWADE allows new building blocks, such as C programs or web-services, through simple annotations in a XML file. SWADE also allows easy insertion of new processing nodes in the system, making the processing capacity scalable. To the end, the machine's owner, connected to the local network or Internet, just needs to install a small packet of web-services that implements the SWADE protocols.

The set of communicating nodes distribute the task of interpreting workflow descriptions; scheduling and executing tasks. The nodes also perform client activities, allowing user interaction with the system, creating and monitoring the execution of experiments; inserting and removal of new processing nodes in the system; among others.

The scheduling job has the collaboration of all nodes, but its main behavior is centralized. One of the nodes assumes the role of a main scheduler, in order to centralize information that allows the balancing while processing multiple workflows. Currently, SWADE presents the FIFO (First In, First Out) scheduling policy, which ensures that the demands are processed at arrival order, according to the availability of the servers. Fig. 5 illustrates this structure.

The implemented language to define workflows allows: sequencing of actions, iteration, conditional execution, parallel execution, passing parameters, variables and asynchronous execution. The system provides further actions for composing workflows. Such actions may vary in nature, for example: an operating system command (Windows or Linux); an executable program (Java or C); third-party tools that can be executed from the command line (MATLAB or Cast3M); web services; among others, with no need of recompiling the system.
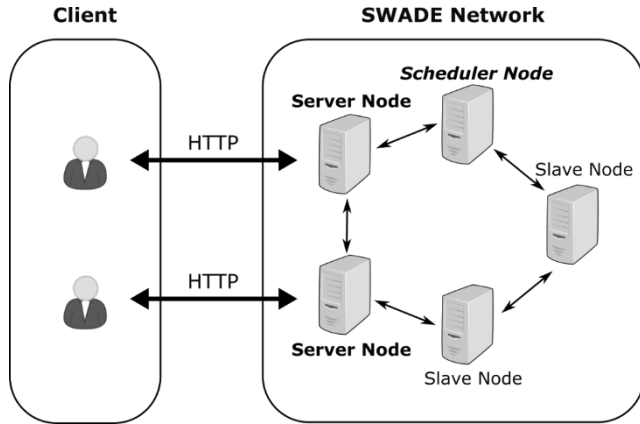
Figure 5. SWADE's Architecture describing your communication network.
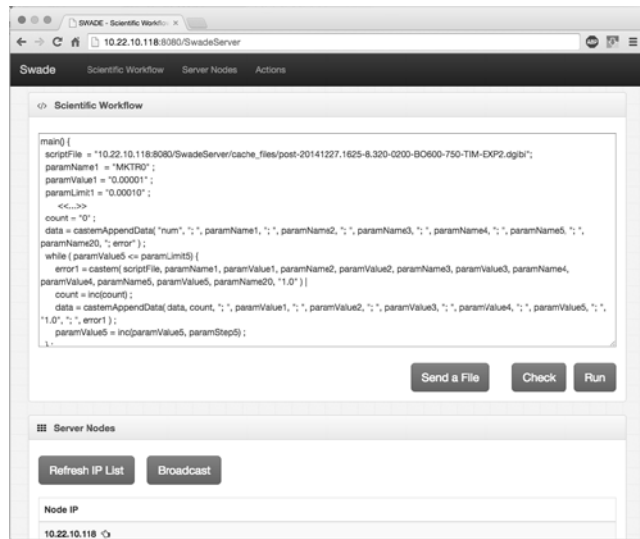Source: The authors.

Figure 6. SWADE client interface with simple script sample.
Source: The authors.

New actions are given a mnemonic name to be used within the workflow description language, and are recognized by SWADE at runtime along with the associated parameters.

The addition of new actions may be managed in a distributed way, by any client node of the system. The technical details of the action activation are encapsulated in XML based configuration files, and loaded only once, at deployment time, making the action available to the user in the form of a function call, such as castem("script.dgibi").

The interface to build, activate and monitor the execution of scientific workflows is shown in Fig. 6, which also illustrates the syntax of SWADE's language. The detailed presentation of the workflow is out of the scope of the present work.

4. Adjustment Strategies

Firstly, the operation of computing simulations was performed manually and took a long period of

experimentation. On these tests, the six characteristic parameters of the material were manually adjusted. Comparison to experimental information resulted in a minimum error of 0.13760, calculated via the Root Mean Square Error (RMSE). The set of manual parameters was: $A_t = 1.0$, $B_t = 5x10^4$, $A_c = 1.5$, $B_c = 1500$, $\varepsilon_{d0} = 10^{-4}$ and $\beta = 1.06$.

Aiming to improve the efficiency and efficacy of the procedure, it was created a systematic adjustment strategy for running Cast3M with varied parameters. Based on laboratory and empirical information, reference [4] states that the six parameters may range according to eq. (6).

A Sensitivity Analysis (SA) [19] was carried out in order to identify the extent at which the Mazars model depends on each input parameter, so as to determine which quantity contributes the most to the output variability and, possibly, requires further research to strengthen the knowledge base and to reduce the output uncertainty.

$$
\begin{aligned}
1.0 &\le A_c \le 1.5 \\
0.7 &\le A_t \le 1.0 \\
10^{-5} &\le \varepsilon_{d0} \le 10^{-4} \\
1000 &\le B_c \le 2000 \\
10^4 &\le B_t \le 10^5 \\
1.0 &\le \beta \le 1.2
\end{aligned}
\tag{6}
$$

Fig. 7 presents the SA for the six input parameters: A_t, B_t, A_c, B_c, ε_{d0} and β. In this study, the lower bound of all parameters was taken as the basis for comparison result. Moreover, the search space of each parameter, comprising the values between the upper and lower bounds, was divided into ten equal parts to be used in the simulations: one parameter was varied at a time while the others remained fixed as the base values.

It can be seen in the Fig. 7 that A_c and B_c were the most influent parameters for the output variability while A_t was the least influencing parameter. Therefore, the A_c and B_c values demand a higher degree of discretization compared to the others, while processing the numerical experiments.
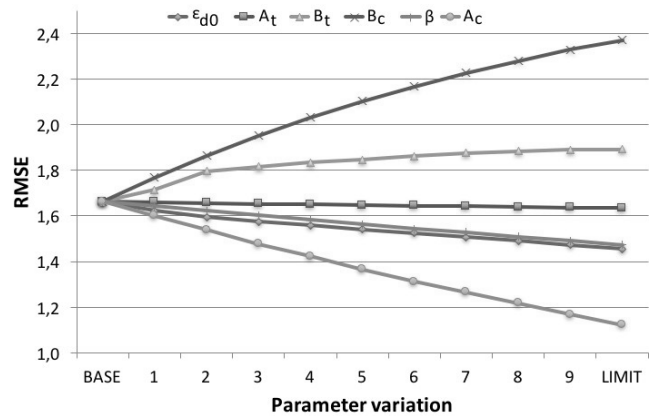

Figure 7. Sensitivity Analysis of input parameters, which contribute the most to output variability of Marzars model.
Source: The authors.

Two approaches were adopted herein. Firstly, the parameters were considered to vary on an arithmetic progression, respecting the intervals given by eq. (6), resulting in 6624 different combinations. After processing, the RMSE was calculated for every set of parameters, by comparison to experimental data. Algorithm 1 presents the logic used in this first strategy.

Secondly, it was assumed broader intervals concerning the six parameters, as given in eq. (7). The aim of this experiment was to verify whether better results would be reached, which could denote the need for further laboratory investigation concerning the material's characteristic parameters. A new strategy was adopted for the adjustment, according to Algorithm 2, consisting in the following steps:

1. six equally spaced values were chosen for the first parameter: the minimum and the maximum bound values and 4 intermediate values;
2. for each value chosen above, it was set a different combination of six parameters;
3. the six combinations were processed simultaneously;
4. after processing, execution errors (RMSE) were evaluated and the set giving in the best fit result is chosen as a reference for the generation of a new set of parameters;
5. If results are improved, repeat the procedure from step 1;
6. If no improvement is observed, repeat the procedure from step 1 - taking the next parameter - or stop the procedure after experimenting with all parameters.

Algorithm 1. Scientific Workflow of the first approach.

begin
    initialize_parameters_value;
    while exists_combination_parameters **do**
        execute_simulations_simultaneously;
        save_rmse;
        increases_parameters_value;
    end_while
end

Source: The authors.

$$1.0 \leq A_c \leq 2.0$$
$$0.2 \leq A_t \leq 1.0$$
$$10^{-5} \leq \varepsilon_{d0} \leq 10^{-3}$$
$$500 \leq B_c \leq 2000 \qquad (7)$$
$$10^2 \leq B_t \leq 10^5$$
$$0.9 \leq \beta \leq 1.5$$

In this second strategy, 294 combinations of parameters were generated to be processed following Algorithm 2.

Algorithm 2: Scientific Workflow of the second approach.

begin
    initialize_parameters_value;
    while exists_better_adjustment **do**
        execute_simulations_simultaneously;
        save_rmse;
        if exists_better_adjustment **then**

            redefine_parameters_value;
        end_if
        set_new_parameters_value;
    end_while
end

Source: The authors.

5. Scientific Workflow Execution

The adopted strategies were mapped to the SWADE language for execution. SWADE was configured to operate with six server nodes - each node in the system presented the following configuration: CPU i7 950 (3.07GHz), 8GB of RAM and network with 100M bit/s.

While mapping the experiment strategy, a number of features of the SWADE language where employed so as to optimize the use of server nodes running each subset of parameters combinations in parallel, according to the number of available processing nodes. This approach, applied to the two strategies, reduced the experimentation time to 4.65% and 0.8% of the sequential execution time (from 27158 to 4102 and 220 minutes, respectively).

Fig. 8 shows the error rates obtained for the 6624 sets of parameters tested in the first strategy. As one can notice in the figure, the error tends to decrease as the sequence of executions progresses, as a result of the applied strategy.

The total processing time demanded for testing all the 6624 combinations of parameters in the first approach was 68 hours and 22 minutes. The smallest RMSE achieved after the execution of the scientific workflow was 0.11882, which is better than the result obtained from them manual adjustment procedure. The set of parameters that gave the best result was: $A_t = 0.7$, $B_t = 10^4$, $A_c = 1.5$, $B_c = 1000$, $\varepsilon_{d0} = 9x10^{-5}$ and $\beta = 1.18$.

The numerical values resulting from the execution of the second strategy are shown in Fig. 9. The total workflow execution time for this approach was 2 hours and 42 minutes - 25 times faster than the previous approach. The best RMSE
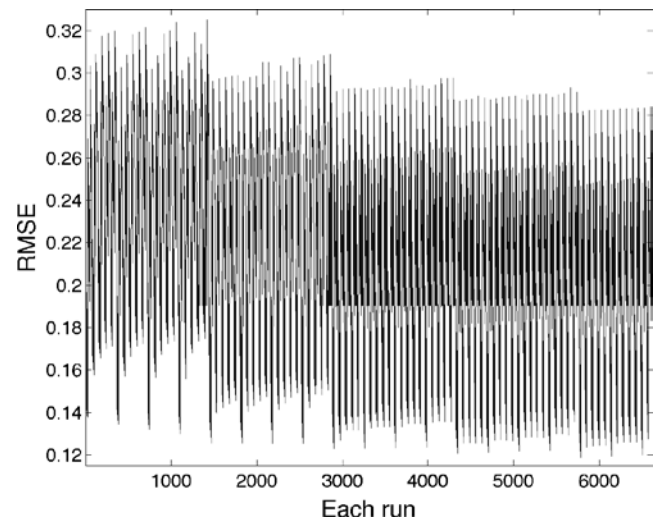

Figure 8. RMSE values obtained for the 6624 sets of parameters tested in the first approach.
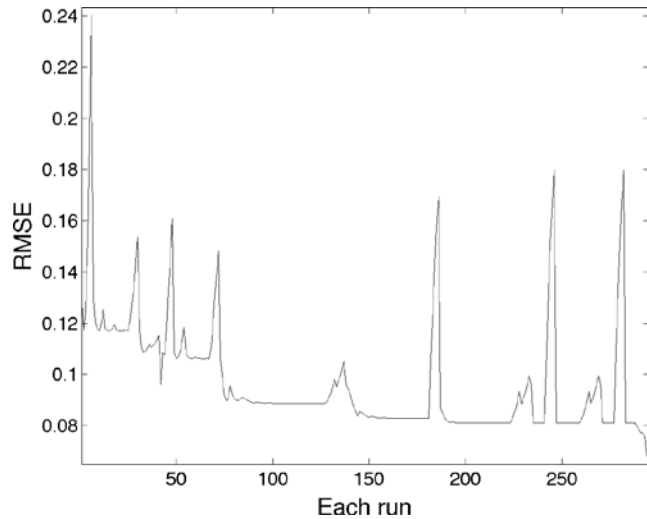Source: The authors.

Figure 9. RMSE values obtained in each execution with set of distinct parameters on second approach.
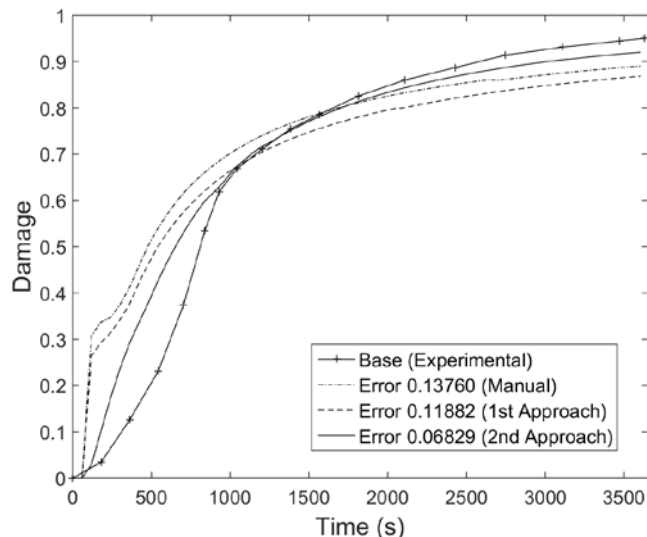Source: The authors.

Figure 10. Curves of damage comparing numerical models to experimental data.
Source: The authors.

found in this experiment was 0.06829, representing an improvement of 1.7 times compared to the previous result. The resulting set of parameters was: $A_t = 0.5$, $B_t = 10^3$, $A_c = 1.932$, $B_c = 1000$, $\varepsilon_{d0} = 3.3 x 10^{-4}$ and $\beta = 1.0$. Considering the limits defined in eq. (6), these procedure results in four parameters assuming values outside the intervals indicated by reference [4]: A_t, B_t A_c, and ε_{d0}.

For comparison purposes, Fig. 10 confronts the experimental damage evolution in time with the curves obtained with the 3 sets of parameters obtained herein. The curves are named by the corresponding RMSE value. As one can see by comparing the 4 curves, the smaller the RMSE value, the better is the curve fitting.

6. Conclusions

The proposed methodology provided remarkable improvement on parameters adjustment, concerning efficiency and efficacy aspects. SWADE contributed on the following aspects of the in silico experimentation life cycle:

- The automation of the experiment execution control avoided the human intervention that is time consuming and error prone;
- The parallel execution of tasks significantly improved execution efficiency;
- The easy integration of third party software into the workflow system, as it was the case with Cast3M. Such task may demand a huge effort on other tools as described in [15].
- SWADE language has a fast learning curve and avoids the need for complex technical details inherent to distributed and heterogeneous systems. Execution of the same task via script languages, such as Python, demands much more effort to a non-expert user.

The adjustment dealt with in the present study may also be treated by means of other kinds of approaches, such as meta-heuristics used in Evolutionary Computing [20]. However, the main contribution herein is to evaluate the applicability of ad-hoc SWS, striving to reach good experimentation results while keeping a compromise with the experimentation process efficiency.

The use of SWADE succeeds to keep a good balance on treating experimentation efficiency at each step of its life cycle. This tool may also be applied to automate experimentations involving other techniques to solve this engineering problem, as mentioned above.

Some issues to be investigated are:

- different approaches to choose the granularity for variation of the model parameters;
- application of new methods for parameter adjustment [21,22] to improve the search strategy;
- improvements on the SWADE scheduling algorithms, such as the one presented in [23],
- refinement of the execution parallelization tasks.

The results indicate that the proposed methodology may evolve to practical real life applications as an auxiliary tool for parameters adjustment and, even, as a means of evaluating the quality of laboratory data.

Acknowledgements

References

[1] Heymann, M., Understanding and misunderstanding computer simulation: The case of atmospheric and climate science - an introduction. Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics, 41(3), pp. 193-200, 2010.

[2] Deelman, E., Gannon, D., Shields, M. and Taylor, I., Workflows and e-science: An overview of workflow system features and capabilities. Future Generation Computer Systems, 25(5), pp. 528-540, 2009.

[3] Gannon, D., Taylor, I.J., Deelman, E. and Shields, M., Workflows for e-Science. Springer- Verlag London Limited, 2007.

[4] Mazars, J., Application de la mécanique de l'endommagement au comportement non linéaire et à la rupture du béton de structure [Application of mechanical damage to the nonlinear behavior and rupture of the concrete structure], PhD Thesis, University Pierre ET Marie Curie, Paris, 1984.

[5] Amaral, R.d.O., Análise computacional termo-mecânica de estruturas de concreto sujeitas a temperaturas elevadas [Thermo-Mechanical Computational Analysis of Concrete Structures Subjected to High Temperatures], MSc. Thesis, Postgraduate Program in Computational Modeling, Federal University of Juiz de Fora - UFJF, Brazil, 2014.

[6] Xing, Z., Influence de la nature minéralogique des granulats sur leur comportement et celui du béton à haute temperature [Influence of the mineralogical nature of the aggregates on their behavior and that of the high-temperature concrete], PhD Thesis, Specialization: Civil Engineering, Cergy-Pontoise University, France, 2011.

[7] Xing, Z., Beaucour, A.L., Hebert, R., Noumowe, A. and Ledesert, B., Influence of the nature of aggregates on the behaviour of concrete subjected to elevated temperature. Cement and concrete research, 41(4), pp. 392-402, 2011.

[8] Ferreira, A.P.G., Farage, M.C., Barbosa, F.S., Noumowé, A. and Renault, N., Thermo-hydric analysis of concrete–rock bilayers under fire conditions. Engineering Structures, 59, pp. 765-775, 2014.

[9] Ferreira, A., Modelagem dos fenômenos de transporte termo-hídricos em meios porosos submetidos a temperaturas elevadas: Aplicação a uma bicamada rocha-concreto, MSc. Thesis, Postgraduate Program in Computational Modeling, Federal University of Juiz de Fora - UFJF, Brazil, 2011.

[10] Le Fichoux, E., Présentation et utilisation de CASTEM, CEA, 2011, 92 P.

[11] Pignotti, E., Edwards, P., Gotts, N. and Polhill, G., Enhancing workflow with a semantic description of scientific intent. Web Semantics: Science, Services and Agents on the World Wide Web, 9(2), 2011.

[12] Seffino, L., Medeiros, C., Rocha, J. and Yi, B., Woodss — A spatial decision support system based on workflows. Decision Support Systems, 27(1-2), pp. 105-123, 1999.

[13] Yu, J. and Buyya, R., A taxonomy of scientific workflow systems for grid computing. SIGMOD Rec., 34(3), pp. 44-49, 2005.

[14] Altintas, I., Barney, O. and Jaeger-Frank, E., Provenance collection support in the kepler scientific workflow system. Springer Berlin Heidelberg, pp. 118-132, 2006. DOI: 10.1007/11890850_14

[15] Bonifácio, A.L.. Análise de ferramentas computadorizadas para suporte à modelagem computacional - Estudo de caso no domínio de dinâmica dos corpos deformáveis [Analysis of computerized tools for computational modeling support – Case study of the dynamics system domain], MSc. Thesis, Postgraduate Program in Computational Modeling, Federal University of Juiz de Fora - UFJF, Brazil, 2008.

[16] Yu, J. and Buyya, R., A taxonomy of workflow management systems for grid computing. Journal of Grid Computing, 3(3-4), pp. 171-200, 2005.

[17] Bavoil, L., Callahan, S.P., Crossno, P.J., Freire, J., Scheidegger, C.E., Silva, C.T. and Vo, H.T., Vistrails: Enabling interactive multiple-view visualizations. In Visualization, 2005. VIS 05. IEEE, pp. 135-142. IEEE, 2005.

[18] Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E.A., Tao, J. and Zhao, Y., Scientific workflow management and the kepler system. Concurrency and Computation: Practice and Experience, 18(10), pp. 1039-1065, 2006.

[19] Iooss, B. and Lemaître, P., A Review on global sensitivity analysis methods. Uncertainty management in simulation-optimization of complex systems. Algorithms and Applications, 59, pp. 101-122, Springer, 2015.DOI: 10.1007/978-1-4899-7547-8_5

[20] Bäck, T. and Schwefel, H.-P., An overview of evolutionary algorithms for parameter optimization. Evolutionary Computation, 1(1), pp. 1-23, 1993. DOI: 10.1162/evco.1993.1.1.1

[21] Qaffou, I., Sadgal, M. and Elfazziki, A., A reinforcement learning method to adjust parameter of texture segmentation. In Informatics and Systems (INFOS), 2010 The 7th International Conference on, pp. 1-5. IEEE, 2010.

[22] Hutter, F., Xu, L., Hoos, H.H. and Leyton-Brown, K., Algorithm runtime prediction: Methods & Evaluation. Artificial Intelligence, 206, pp. 79-111, 2014.

[23] Soares, S.S.R.F., Gonçalves, L.B. and de Lyra, A.R., Um algoritmo genético para o problema da sequência mais próxima [A genetic algorithm for the problem of the closest sequence]. XLV Simpósio Brasileiro de Pesquisa Operacional, Brazil, 2013.

A. Lage-Bonifácio, is a PhD. student in the Postgraduate Program in Computational Modeling in the Federal University of Juiz de Fora - UFJF, Brazil. He received his BS from Centro de Ensino Superior de Juiz de Fora in 2002 and MSc from UFJF in 2008. His research interests include computational modeling of concrete structures, computational intelligence, machine learning, scientific workflows and distributed programming.
ORCID: 0000-0001-5230-4311

R. de Oliveira-Amaral, is a MSc. in the Postgraduate Program in Computational Modeling in the Federal University of Juiz de Fora - UFJF, Brazil. She received his BS from UFJF in 2011.
ORCID: 0000-0003-2717-744X

C. de Barros-Barbosa, is an associate professor and research engineer at the Federal University of Juiz de Fora - UFJF, Brazil. He received her BSc. from University of Estado do Rio de Janeiro - UERJ in 1989, MSc from Federal University of São Carlos - UFSCAR in 1995 and PhD. from University of Twente, Holanda in 2001. His research interests include distributed systems, model-driven system development, scientific workflows and e-science.
ORCID: 0000-0002-4701-603X

M.C. Resende-Farage, is an associate professor and research engineer at the Federal University of Juiz de Fora - UFJF, Brazil. She received her BSc from UFJF in 1991, MSc from Federal University of Rio de Janeiro – UFRJ in 1995 and PhD. from UFRJ in 2000. Her research interests include experimental analysis and computational modeling of concrete structures.
ORCID: 0000-0002-5344-0937

F. de Souza-Barbosa, is an associate professor and research engineer at the Federal University of Juiz de Fora - UFJF, Brazil. He received his BSc. from UFJF in 1994, MSc from Federal University of Rio de Janeiro – UFRJ in 1996 and PhD. from UFRJ in 2000. He is member of the Editorial Board of the IABSE Journal – Structural Engineering International (SEI). His research interests include structural dynamics, concrete structures and computational modeling of structures.
ORCID: 0000-0002-7991-8425