DOI: 10.1111/mice.12762

ORIGINAL ARTICLE

COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING WILEY

Adaptive approximate Bayesian computation by subset simulation for structural model calibration

José Barros[1,2] | **Manuel Chiachío**[2,3] | **Juan Chiachío**[2,3] | **Frank Cabanilla**[1]

[1] Faculty of Engineering, Catholic University of Santiago de Guayaquil, Guayaquil, Ecuador

[2] Department of Structural Mechanics and Hydraulics Engineering, University of Granada, Granada, Spain

[3] Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI), Granada, Spain

Correspondence
Manuel Chiachío, Department of Structural Mechanics and Hydraulics Engineering, 18011 Granada, Spain.
Email: mchiachio@ugr.es

Abstract

This paper provides a new approximate Bayesian computation (ABC) algorithm with reduced hyper-parameter scaling and its application to nonlinear structural model calibration problems. The algorithm initially takes the ABC-SubSim algorithm structure and sequentially estimates the algorithm hyper-parameter by autonomous adaptation following a Markov chain approach, thus avoiding the error associated to modeler's choice for these hyper-parameters. The resulting algorithm, named $\mathcal{A}^2$BC-SubSim, simplifies the application of ABC-SubSim method for new users while ensuring better measure of accuracy in the posterior distribution and improved computational efficiency. A first numerical application example is provided for illustration purposes and to provide a comparative and sensitivity analysis of the algorithm with respect to initial ABC-SubSim algorithm. Moreover, the efficiency of the method is demonstrated in two nonlinear structural calibration case studies where the $\mathcal{A}^2$BC-SubSim is used as a tool to infer structural parameters with quantified uncertainty based on test data. The results confirm the suitability of the method to tackle with a real-life damage parameter inference and its superiority in relation to the original ABC-SubSim.

1 | INTRODUCTION

Bayesian methods are nowadays at the core of development in numerous fields of science and engineering, probably due to the maturity of the statistical computing tools in conjunction with the rampant increase of computational power. In structural engineering, the Bayesian approach provides a powerful and robust framework to system identification (Jiang et al., 2007; Perez-Ramirez et al., 2019; Yuen et al., 2019), comprising either model parameter updating or model class selection (Ching et al., 2006; Sharma et al., 2015; Yin & Zhu, 2020; Zhang et al., 2013).

Irrespectively, closed form solutions to the Bayesian inverse problem (BIP) are rarely available due to the complexity of the multidimensional integrals involved or due to difficulty in the formulation of a likelihood function Beck (2010). Hence a branch of approximation methods have been explored in the literature during decades to provide efficient and effective solutions to practical applications of the BIP. These methods mostly involve variational methods and stochastic simulation approaches (Stuart, 2010).

Apart from these methods, approximate Bayesian computation (ABC) methods emerged with the purpose of solving the BIP in those cases where the evaluation of the *likelihood function* is computational prohibitive or analytically intractable (Marjoram et al., 2003; Tavaré et al., 1997). ABC methods have been under continuous development since the last two decades, and a number of ABC variants have been proposed in the literature by joining the ABC principles with variational inference or Monte Carlo methods like the Kernel ABC (Park et al., 2016), Lazy ABC

(Prangle, 2016), Coupled ABC (Neal, 2012), Empirical-likelihood ABC (Mengersen et al., 2013), or Bootstrap-likelihood ABC (Zhu et al., 2016). Other variants combine the ABC approach with efficient sampling algorithms like the ABC-Partial Rejection Control (Sisson et al., 2007), ABC-Sequential Monte Carlo (Del Moral et al., 2012; Toni et al., 2009), ABC-Particle Monte Carlo (Beaumont et al., 2009), ABC-Parallel Tempering (Baragatti et al., 2013), ABC by Simulated Annealing (Albert et al., 2015), and ABC by Subset Simulation (Chiachio et al., 2014). The reader is referred to Sisson et al. (2018) for a comprehensive overview of the ABC methods and to Karabatsos et al. (2018) for a recent comparative review.

The majority of these ABC variants have been proposed in the form of new ABC algorithms that have been successfully used for model inference and calibration in a wide range of application fields, such as molecular dynamics (Dutta et al., 2018; Kulakova, 2017), biology (Bianconi et al., 2019), hydrology (Kavetski et al., 2018), health sciences (Da Costa et al., 2018; McKinley et al., 2018; Rutter et al., 2019), environmental radioactivity (Nishina et al., 2018), communications (Bharti & Pedersen, 2019), and physics (Christopher et al., 2018). Engineering applications (Sala & Soriguera, 2020), and particularly structural engineering applications, have also received attention from the ABC community mainly to infer unknown structural performance parameters from nonlinear models (Ben Abdessalem et al., 2019; Betz, 2017; De et al., 2019; Lam et al., 2018; Song et al., 2019; Tiboaca, 2016).

Despite their successful contribution for solving complex problems in a wide range of applications, the ABC variants available in the literature require the definition of a number of sensitive algorithm hyper-parameters aparat from the ABC scaling parameters, as will be shown next, in Section 2.1. that highly influence the efficiency of the algorithm and also the quality of the solution. This drawback puts some extra difficulties for extending these algorithms to real-world engineering problems as previous expertise is required to tune these parameters.

In this paper, a new self-adaptive approximate Bayesian computation algorithm is presented. The algorithm has been named *Adaptive ABC by Subset Simulation* (also referred to as $\mathcal{A}^2$BC-SubSim), and uses a Markov Chain approach to adaptively and autonomously scale the algorithm hyper-parameters, thus avoiding the modeler's intervention. The proposed algorithm takes the basic structure of the ABC-SubSim algorithm (Chiachio et al., 2014), as it has been proven to be one of the most efficient ABC algorithms in the literature and also because it is included in several well-known ABC user-platforms like ABCpy Dutta et al. (2017) and Pi4U (https://github.com/cselab/pi4u). The resulting $\mathcal{A}^2$BC-SubSim is a general purpose algorithm for Bayesian inference, however, here the algorithm is illustrated as a tool for autonomous parameter calibration of complex structural models based on real-world test results.

Indeed, after illustration using a simple numerical example, a real-world engineering case study about a well-known nonlinear model calibration of a concrete column under cycling lateral loads is presented to demonstrate the full potential of the $\mathcal{A}^2$BC-SubSim algorithm. The performance of the algorithm is further demonstrated in a calibration exercise of a complex dynamical model from a tall-building subjected to seismic excitation and using structural health monitoring data. This exercise is provided to show the capacity of $\mathcal{A}^2$BC-SubSim in a large and complex application that includes multiple (>30) uncertain parameters.

Structural model calibration is a common practice in engineering to adapt an analytical or semi-analytical structural model to a specific real-world application scenario based on test data, and constitutes the basis for the formulation of the majority of structural design codes and standards. Recent examples of structural model calibration can be found in Lee and Han (2018) and LeBorgne and Ghannoum (2014), who adopted a laboratory-based *by-hand* procedure to calibrate nonlinear models for concrete columns. A similar calibration procedure was presented in Haselton et al. (2008), which stated that degradation parameters were empirically obtained by trial and error (following a standardized procedure) until a good *visual match* between the model and the experimental results were achieved. Also in Haselton et al. (2008), a regression analysis was carried out and empirical equations were proposed for model parameter estimation after individual calibrations of more than 200 column tests.

The aforementioned works evidence that manual calibration and regression analyses of nonlinear structural model lacks of rigor and might imply biased results even when highly trained structural specialists were involved. To the best of the authors' knowledge, the calibration of complex structural models has never been done before using heterogeneous sources of data. The latter, along with the difficulties for defining and computing the likelihood function, makes the ABC approach adequate to overcome these limitations and provide a bone fide structural model calibration with quantified uncertainty. More specifically, the results shown in this paper after the application of $\mathcal{A}^2$BC-SubSim demonstrate that the nonlinear hysteretic column model can be efficiently calibrated considering uncertainty about the parameters, the model, and the data with minimal algorithm manipulation. In application to the tall-building model calibration, the results show that the proposed algorithm is able to perform model inference of large structural models with a feasible computational cost, and requiring minimal algorithm tuning.

The paper is organized as follows. In Section 2, the adaptive approximate Bayesian computation by Subset Simulation ($\mathcal{A}^2$BC-SubSim) algorithm is presented. In Section 2.3, a simple but illustrative example is shown to validate the implementation. A structural nonlinear model of a reinforced concrete column subjected to cyclic degradation is presented in Section 3 along with its parameter inference using the $\mathcal{A}^2$BC-SubSim algorithm. In Section 4, a comparison of the proposed inference-based calibration method with conventional by-hand procedure is presented, suggestions to get consistency on the inferred parameters are given, and the extension to a large structural model is presented. Finally, in Section 5 some concluding remarks are provided.

2 | ADAPTIVE APPROXIMATE BAYESIAN COMPUTATION BY SUBSET SIMULATION

The $\mathcal{A}^2$BC-SubSim algorithm proposed herein is based on the structure of the ABC-SubSim method. Thus a concise description of ABC-SubSim is provided here to confer the reader with the basis about the method under a unified notation.

2.1 | Bayesian model updating by ABC-SuBSim

The focus of Bayesian model updating is to update the prior information about the value of a set of uncertain model parameters $\boldsymbol{\theta} \in \boldsymbol{\Theta} \subseteq \mathbb{R}^{n_p}$ from a parameterized model $\mathbf{x} = g(\mathbf{u}, \boldsymbol{\theta}) \in \mathcal{X} \subset \mathbb{R}^d$ based on the information given by the data $\mathbf{y} \in \mathcal{D} \subset \mathbb{R}^d$, where $g : \mathbb{R}^{n_u \times n_p} \to \mathbb{R}^d$, $\mathbf{u} \in \mathbb{R}^{n_u}$ are input values to g, and $\mathcal{D}$ is the *observation space*. Following the Bayesian formulation, the solution is not a single value of $\boldsymbol{\theta}$; on the contrary, Bayes' Theorem takes the initial quantification of the plausibility of $\boldsymbol{\theta}$, which is expressed by the *prior* PDF $p(\boldsymbol{\theta})$, and updates this plausibility using the information in the data set $\mathcal{D}$ through the *likelihood function* $p(\mathbf{y}|\mathbf{x})$ to obtain the *posterior* PDF of the state variable $p(\boldsymbol{\theta}|\mathbf{y})$. The interested reader is referred to Beck (2010) for further information about Bayesian model updating.

However, there are situations where the likelihood function is unknown or analytically intractable, for which the ABC methods (Marjoram et al., 2003), also known as *likelihood-free computation algorithms*, provide an efficient alternative. These methods bypass the evaluation of the likelihood function using a simulation-based approach. Through a specific tolerance parameter ξ, the method selects as posterior samples the pairs $(\mathbf{x}, \boldsymbol{\theta}) \in \mathcal{H} \subseteq \mathcal{X} \times \boldsymbol{\Theta}$ such that the model simulations $\mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ lay within a specific region around $\mathbf{y}$, namely $\mathcal{B}_\xi(\mathbf{y}) = \{\mathbf{x} \in \mathcal{D} : \rho(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}) \leqslant \xi\}$, where $\rho(\cdot) : \mathbb{R}^{d \times n_p} \to \mathbb{R}$ is a user-defined metric function used to measure the closeness of the simulated output $\mathbf{x}$ to the data $\mathbf{y}$. By this means, the ABC marginal posterior of parameters is expressed as $p_\xi(\boldsymbol{\theta}) \propto P(\mathbf{x} \in \mathcal{B}_\xi(\mathbf{y})|\boldsymbol{\theta})p(\boldsymbol{\theta})$, where $P(\mathbf{x} \in \mathcal{B}_\xi(\mathbf{y})|\boldsymbol{\theta})$ assigns the unity when $\rho(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}) \leqslant \xi$, and 0 otherwise.

ABC-SubSim (Chiachio et al., 2014) exploits the efficient simulation framework of Subset Simulation, originally proposed by Au and Beck (2001) as an estimator of small failure probabilities. By Subset simulation, a rare event simulation problem is transformed into the product of a series of simulations with larger probabilities whereby the computational effort is reduced. Indeed, in ABC-SubSim, the region $\mathcal{H}$ of possible solutions is assumed to be defined as the intersection of m nested regions in the $\mathcal{X} \times \boldsymbol{\Theta}$ space, that is, $\mathcal{H}_1 \supseteq ... \mathcal{H}_j \supseteq ... \supseteq \mathcal{H}_m = \mathcal{H}$, where

$$\mathcal{H}_j = \{(\mathbf{x}, \boldsymbol{\theta}) : \rho(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y}) \leqslant \xi_j\} \tag{1}$$

In the last equation, the tolerance values follow a decreasing sequence $\xi_1 \geqslant \xi_2 ... \geqslant \xi_m$ whose values are adaptively determined such that the sample estimate $\bar{P}_j \triangleq P(\mathcal{H}_j|\mathcal{H}_{j-1})$ satisfies $\bar{P}_j = p_0$, with $P(\mathcal{H}_j) = P((\mathbf{x}, \boldsymbol{\theta}) \in \mathcal{H}_j)$, and p_0 is a conditional probability acting as algorithm hyper-parameter defined by the modeler.

2.2 | The $\mathcal{A}^2$BC-SubSim algorithm

As seen above, the conditional probability p_0 in ABC-SubSim is a hyper-parameter that needs to be fixed in advance. As shown in Chiachio et al. (2014), p_0 has a strong influence on the computational burden of the algorithm along with the quality of the ABC posterior approximation. The *$\mathcal{A}^2$BC-SubSim* method avoids the manual tuning of p_0 by numerical adaptation, as shown next.

First, let us consider that $\bar{P}_j$ is not fixed to a specific value p_0 and that the sequence $\bar{P}_1, \bar{P}_2, ..., \bar{P}_j, ..., \bar{P}_m$ follows a Markov chain whose initial state $\bar{P}_1 = p_1$ is known, while the rest $\bar{P}_j = p_j, j \geqslant 2$, are randomly chosen from a truncated normal PDF given by

$$\mathcal{N}^*(a, b, \mu, \sigma) = \frac{1}{\sigma} \frac{\phi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)} \tag{2}$$

where a and b are the lower and upper truncation values, μ and σ are the mean and standard deviation, respectively, $\phi(\cdot)$ is the standard normal PDF, and $\Phi(\cdot)$ is its cumulative distribution function.

Next, let us consider that a sample of K values $p_j^{(1)}, p_j^{(2)}, \ldots, p_j^{(k)}, \ldots, p_j^{(K)}$, distributed following Equation (2), are available as possible candidates for any subset $j \geqslant 2$. To determine the best choice for p_j, let us now introduce a weighting function $\omega_k : \mathbb{R} \to [0,1]$, which assigns a value within the $[0,1]$ interval to each sample $p_j^{(k)}$, so that the chosen $p_j = p_j^{(k)}$ is selected with probability ω_k among the K samples. This weighting function is defined based on the following criteria:

- The expected acceptance ratio of the Markov chain Monte Carlo (MCMC) algorithm, α, which should be as near as possible to an optimum value;
- The selected p_j, which should preferably produce an associate tolerance value ξ_j as close as possible to the final tolerance, ξ_f.

The first criterion is based on Papaioannou et al.'s (2015) observations on the optimum acceptance ratio of an MCMC algorithm, where $\alpha \approx 0.40$ is recommended to maximize the efficiency of the Subset Simulation method; note that this allows the modeler to avoid some required preruns to set a hyper-parameter for the original ABC-SubSim (Chiachio et al., 2014), like the variance in the proposal PDF. The second criterion is proposed to minimize the amount of subsets required to reach the final tolerance ξ_f, thus increasing the computational efficiency. Therefore, the weighting function can be mathematically expressed as

$$\omega_k = u_k \cdot \upsilon_k \tag{3}$$

where u_k and υ_k are ad hoc functions to take into account the first and second aforementioned criteria, respectively. The u_k factor can be any concave downward function whose maximum is located near the recommended value 0.40. In this work, the following expression is adopted for u_k:

$$u_k = \exp\left(-\frac{(\tilde{\alpha}_k - 0.40)^2}{2\sigma_\alpha^2}\right) \tag{4}$$

where $\tilde{\alpha}_k$ is the sample estimate acceptance ratio when $p_j^{(k)}$ is adopted, and σ_α is a factor to set the influence of u_k in ω_k. Further insight about the influence of σ_α on the computational efficiency is provided in Section 2.4.

For the second factor υ_k, a distance function is proposed as follows:

$$\upsilon_k = 1 - \frac{\xi_j^{(k)} - \xi_{\min}}{\xi_{\max} - \xi_{\min}} \tag{5}$$

where $\xi_{\min}$ and $\xi_{\max}$ represent the minimum and maximum metric values obtained for the $\{p_j^{(k)}\}_{k=1}^K$ samples. Note that Equation (5) makes unity when the intermediate tolerance $\xi_j^{(k)}$ associated to a trial $p_j^{(k)}$ equals $\xi_{\min}$ and zero in the opposite case. Note that the weighting function ω_k conveys a healthy balance between simulation efficiency and computational cost, which adaptively penalizes the p_j values whose associated u_k or υ_k are low and favor those whose balance is high. This observation is further discussed in Section 2.4, specifically in Figure 3c.

Finally, note that the choice of $p_j^{(k)}$ constitutes a stochastic random process and, therefore, the shape of u_k and υ_k functions will not have significant influence on the behavior of the algorithm, as long as their values lay within those proposed in Section 2.3. For instance, taking a different weighting function u_k^*, like the one in Equation (6), leads to similar results as denoted in Figure 3c, when comparing u_k^* to $\sigma_\alpha = 0.10$:

$$u_k^* = \begin{cases} 5\tilde{\alpha}_k - 1 & \text{if } 0.2 < \tilde{\alpha}_k \leq 0.4 \\ 3 - 5\tilde{\alpha}_k & \text{if } 0.4 < \tilde{\alpha}_k \leq 0.6 \\ 0.01 & \text{otherwise} \end{cases} \tag{6}$$

An algorithmic description of $\mathcal{A}^2$BC-SubSim is presented as Algorithm 1. The adaptive selection process of p_j values is shown in steps 8–22. Note that the hyper-parameter K is used to set the amount of p_j candidates to be evaluated in every subset, whereas n_k sets the algorithm runs needed to estimate the functions u_k and υ_k. To alleviate computational cost, the algorithm is implemented so that similar p_j candidates (say, with less than 1% difference) are evaluated only once. A flowchart of the proposed algorithm is presented in Figure 1, which also depicts how the ABC-SubSim steps are integrated into the $\mathcal{A}^2$BC-SubSim.

2.3 | Illustrative example

Let us consider a cantilever prismatic column with 0.4-m square cross section, 2-m length that is loaded at the top with a force $F = 1$ kN. For the sake of illustration, let us also consider that the structural material degrades at a unknown constant rate ζ affecting the Young modulus by reducing it from an initial value $E_0 = 40$ MPa following the exponential function:

$$x_n = e^{-\zeta} x_{n-1} + \upsilon_n \tag{7}$$

where subscript $n \in \mathbb{N}$ denotes time in weeks, x refers to the Young's modulus prediction, and υ is a model error term, which is assumed to follow a Gaussian distribution

Algorithm 1 $\mathcal{A}^2$BC-SubSim algorithm

Input: N {number of samples per intermediate level}, n_p {size of the model parameters θ vector}, K {number of trial samples of p_j}, n_k {fraction of N to define the amount of MCMC chains in each trial}, ξ_f {final tolerance value}, m {maximum number of simulation levels allowed}, $\bar{P}_1 = p_1 \in [0,1]$ {Unconditional probability from first simulation level. Can be fixed to 1 }, σ_α, {influence parameter from Eq. (4) }.

Output: $\theta = \left(\theta_m^{(1)}, \mathbf{x}_m^{(1)}\right), \ldots, \left(\theta_m^{(N)}, \mathbf{x}_m^{(N)}\right)$

Algorithm

1: Sample $\left(\theta_1^{(1)}, \mathbf{x}_1^{(1)}\right), \ldots \left(\theta_1^{(N)}, \mathbf{x}_1^{(N)}\right)$, where $(\theta, \mathbf{x}) \sim p(\mathbf{x}|\theta)p(\theta)$

Subset Simulations

2: **for** $j = 1, \ldots, m$ **do**

3: **for** $n = 1, \ldots, N$ **do**

4: Evaluate $\rho_j^{(n)} = \rho\left(\theta_{j-1}^{(n)}, \mathbf{x}_{j-1}^{(n)}, \mathcal{D}\right)$

5: **end for**

6: Renumber $\left[\left(\theta_{j-1}^{(n)}, \mathbf{x}_{j-1}^{(n)}\right), n : 1, \ldots, N\right]$ so that $\rho_j^{(1)} \leq \rho_j^{(2)} \leq \ldots \leq \rho_j^{(N)}$

7: Set $\xi_{\min} = \rho_j^{(1)}$; $\xi_{\max} = \rho_j^{(N)}$

Adaptively select p_j

8: Sample $\{p_j^{(k)}\}_{k=1}^K$, **if** $j = 1$, from: $\mathcal{U}\,(a = 0.1, b = 0.5)$, and from: $\hat{\mathcal{N}}\left(a = 0.1, b = 0.5, \mu_{p_{j-1}}, \sigma_{p_{j-1}}\right)$, **otherwise**

9: **for** all k **do**

10: Set $\xi_j^{(k)} = \frac{1}{2}\left(\rho_j^{Np_j^{(k)}} + \rho_j^{Np_j^{(k)}+1}\right)$

11: Randomly select Nn_k seeds $\left(\theta_{j-1}^{(1)}, \mathbf{x}_{j-1}^{(1)}\right), \ldots, \left(\theta_{j-1}^{(Nn_k)}, \mathbf{x}_{j-1}^{(Nn_k)}\right)$

12: Set $q = 0$ (Auxiliary variable)

13: **for** $i = 1, \ldots, Nn_k$ **do**

14: Generate `Ceil`$[1/p_j^{(k)}]$ states of a Markov chain from the seed $\left(\theta_{j-1}^{(i)}, \mathbf{x}_{j-1}^{(i)}\right)$ (e.g. using Modified Metropolis Algorithm (MMA) Au & Beck (2001)): $\left[\left(\theta_j^{(i),s}, \mathbf{x}_j^{(i),s}\right)\right]_{s=1}^{1/p_j^{(k)}}$

15: Evaluate the metric $\rho_j^{(i),s} = \rho\left(\theta_j^{(i),s}, \mathbf{x}_j^{(i),s}, \mathcal{D}\right)$

16: **if** $\rho_j^{(i),n_k} <= \xi_j^{(k)}$ **then**

17: $q \leftarrow q + 1$

18: **end if**

19: **end for**

20: Calculate the acceptance ratio as $\tilde{\alpha}_k = \dfrac{q}{n_k N}$

21: Evaluate u_k, v_k, and ω_k (use Eqs. (4), (5) and (3), respectively)

22: Set $p_j = p_j^{(k)}$ with probability min $= \{1, \omega_k\}$

23: **end for**

24: Evaluate $\mu_{p_j} = \dfrac{\sum_{k=1}^{K} p_j^{(k)}\omega_k}{\sum_{k=1}^{K}\omega_k}$ and $\sigma_{p_j} = \sqrt{\dfrac{\sum_{k=1}^{K}\left(p_j^{(k)} - \mu_{p_j}\right)^2}{K}}$ for each pair of $\left(p_j^{(k)}, \omega_k\right)$ (used further in step 8 for next j)

25: Fix $\xi_j = \frac{1}{2}\left(\rho_j^{Np_j} + \rho_j^{Np_j+1}\right)$

26: **for** $\ell = 1, \ldots, Np_j$ **do**

27: Select as seed $\left(\theta_j^{(\ell),1}, \mathbf{x}_j^{(\ell),1}\right) = \left(\theta_{j-1}^{(\ell)}, \mathbf{x}_{j-1}^{(\ell)}\right) \in \mathcal{H}_j$

28: Run MMA to generate `Ceil`$[1/p_j]$ Markov chain states lying in $\mathcal{H}_j$: $\left[\left(\theta_j^{(\ell),s}, \mathbf{x}_j^{(\ell),s}\right)\right]_{s=1}^{1/p_j}$

29: **end for**

30: Renumber $\left[\left(\theta_j^{(\ell),s}, \mathbf{x}_j^{(\ell),s}\right) : \ell = 1, \ldots, Np_j; s = 1, \ldots, 1/p_j\right]$ as $\left[\left(\theta_j^{(1)}, \mathbf{x}_j^{(1)}\right), \ldots, \left(\theta_j^{(N)}, \mathbf{x}_j^{(N)}\right)\right]$

31: **if** $\xi_j \leq \xi_f$ **then**

32: End algorithm

33: **end if**

34: **end for**

with zero mean and unknown standard deviation σ, that is, $v_n \sim \mathcal{N}(0, \sigma)$. Also, a sensor is assumed to be available at the top of the column to measure weekly deflections δ_n following a measurement equation, as follows:

$$y_n = h(x_n) + \psi_n \tag{8}$$

where y_n refers to the measured deflection, δ_n and ψ_n denote the measurement error term, which is assumed as a Gaussian zero-mean distribution with a known standard deviation, that is, $\psi_n \sim \mathcal{N}(0, \sigma)$ where $\sigma_\psi = 10^{-6}$, expressed in meters units. In Equation (8), the function $h : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ and can be expressed from elasticity

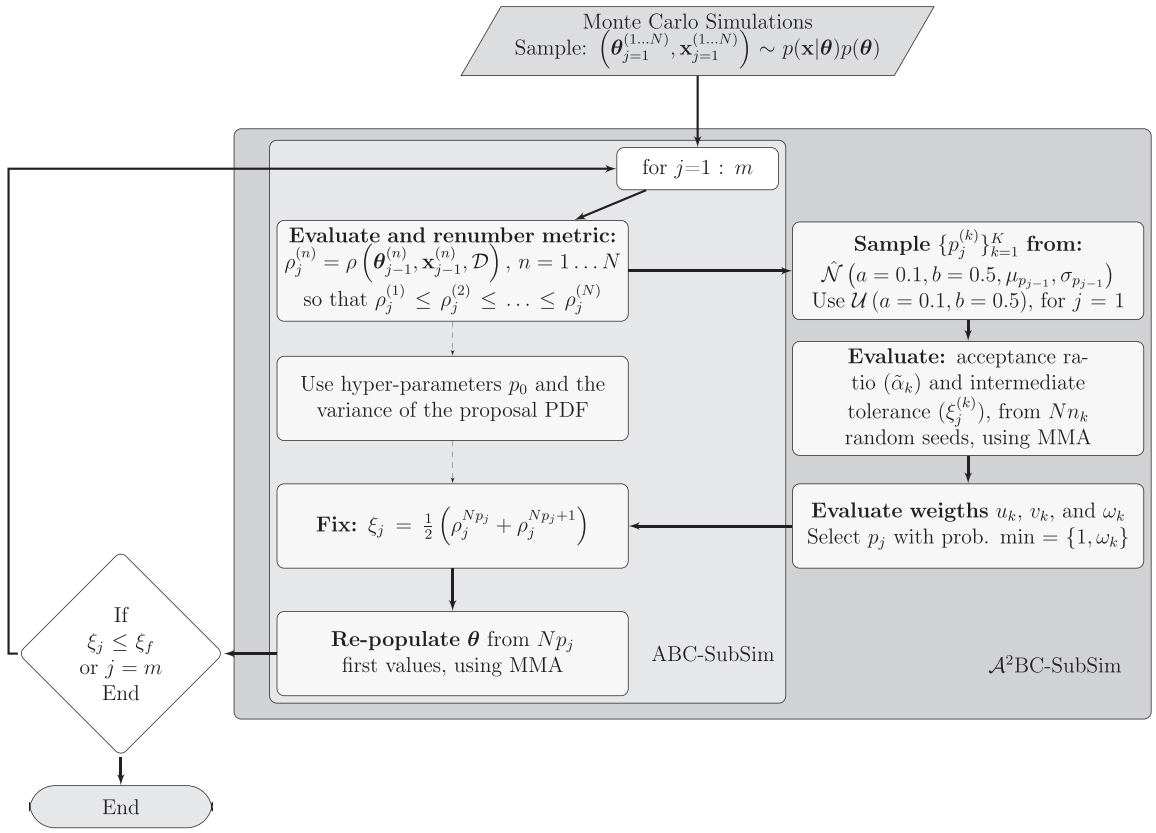


FIGURE 1 $\mathcal{A}^2$BC-SubSim algorithm flowchart. Dashed arrows show the path of the original ABC-SubSim whilst solid ones display the proposed parameter auto-tuning steps

theory as $h = \frac{FL^3}{3x_n I}$, where I is the inertia momentum of the cross section.

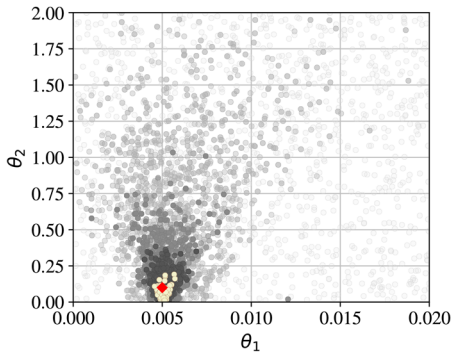
In this example, the degradation rate and the standard deviations of the model error term are selected as unknown model parameters, so that $\boldsymbol{\theta} = \{\theta_1, \theta_2\} = \{\zeta, \sigma\}$. The uniform PDFs $p(\theta_1) = \mathcal{U}[0.0001, 0.02]$ and $p(\theta_2) = \mathcal{U}[0.01, 2]$, respectively, are considered as prior PDFs for the model parameters. The data for this example are synthetically generated from Equations (7) and (8), considering $\boldsymbol{\theta}_{\text{true}} = (0.005, 0.1)$ for a time period of 200 [weeks], that is, $\mathcal{D} = \{\delta_n\}_{n=0}^{200}$ as shown in Figure 2c and d (refer to the blue plot).

In this exercise, the ABC-SubSim and the $\mathcal{A}^2$BC-SubSim algorithms are comparatively used to estimate the approximate posterior $p_\xi(\boldsymbol{\theta}|\mathcal{D})$, with $\xi_f = 80$ [MPa], $N = 5000$ (amount of samples per intermediate level), and using a $\mathbb{L}_1-$norm as metric function, that is, $\rho_{(\mathbf{x},\boldsymbol{\theta},\mathcal{D})} = \sum_{n=0}^{200} \|X_n - x_n\|$, where X_n is the Young modulus obtained from Equation (7) using $\boldsymbol{\theta}_{\text{true}}$, acting as measured Young modulus during the 200 weeks period. The ABC-SubSim algorithm is used with $p_0 = 0.2$, whereas the $\mathcal{A}^2$BC-SubSim is scaled using $\sigma_\alpha = 0.1$, $K = 3$, and $n_k = 0.02$, according to the suggestions given in Section 2.4.

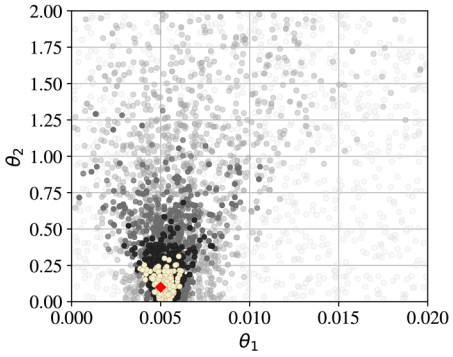
The ABC-SubSim and the $\mathcal{A}^2$BC-SubSim results are presented in Figure 2. In panels (a) and (b), circles represent samples in the model parameter space, whereas the brighter gray circles correspond to prior samples. To reveal the uncertainty reduction, the intermediate posterior is superimposed in increasing gray tones. The results show that the approximate posterior samples (in yellow) are close to the $\boldsymbol{\theta}_{\text{true}}$ in both cases. Panels (c)–(f) provide comparative analysis in terms of model accuracy and cumulative error with respect to the data. In this numerical example, the ABC-SubSim required $m = 5$ simulation levels with 25,000 model evaluations to reach the desired tolerance, whereas $\mathcal{A}^2$BC-SubSim employed 17,016 model evaluations and $m = 3$ simulation levels. Observe that the $\mathcal{A}^2$BC-SubSim posterior samples are substantially closer to $\boldsymbol{\theta}_{\text{true}}$, hence better model response can be obtained if posterior samples from $\mathcal{A}^2$BC-SubSim are used to reproduce the model. This aspect will be further corroborated in Section 3 within the context of an engineering case study.

2.4 | Hyper-parameters evaluation

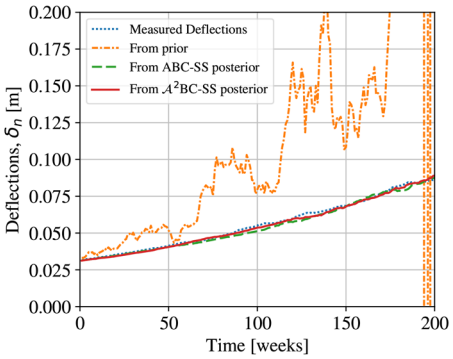
The $\mathcal{A}^2$BC-SubSim is presented in this paper as a variant to the original ABC-SubSim algorithm to circumvent the need of manually scaling the hyper-parameter p_0; however, Section 2 has shown that new control parameters are required, namely K, n_κ, and σ_α. In this



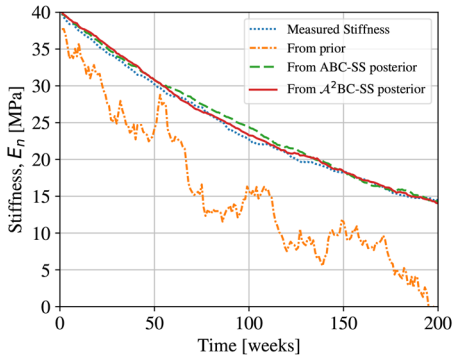
(a) $\mathcal{A}^2$BC-SubSim posterior output



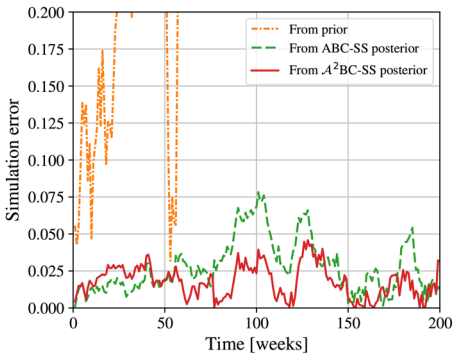
(b) ABC-SubSim posterior output



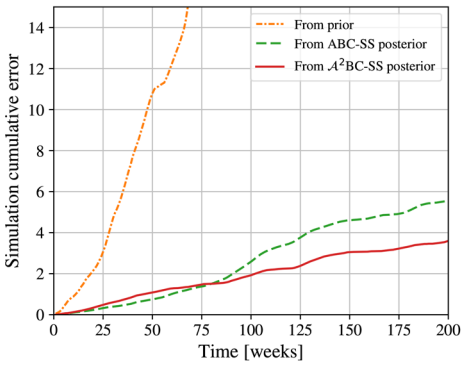
(c) Weekly evolution of column deflections



(d) Weekly evolution of Young modulus



(e) Evolution of the error of simulated deflections



(f) Acumulated error of simulated deflections

FIGURE 2 Comparative results after the application of $\mathcal{A}^2$BC-SubSim and ABC-SubSim algorithms to the cantilever column example from Section 2.3. Panels (a) and (b) show scatter plots of posterior samples of $\boldsymbol{\theta}$ for intermediate levels, the final level (in yellow) and the true values (in red). Panels (c) and (d) represent evaluation of model response using the *Maximum a Posteriori* (MAP) values of the inferred model parameters $\boldsymbol{\theta}$. Panels (e) and (f) represent the comparative error on the simulation of the deflections

section, a sensitivity analysis is presented to show the influence of the aforementioned $\mathcal{A}^2$BC-SubSim control parameters whereby recommendations to fix their values can be obtained.

To this end, let us start by investigating the influence of K, which sets out the amount of p_j trial samples produced at the jth simulation level. Figure 3a shows the statistics taken from 100 independent runs of the algorithm using $K = \{3, 5, 10\}$, whereas the rest of hyper-parameters are fixed to $\sigma_\alpha = 0.1$, $n_\kappa = 0.04$, and $N = 2000$. The results are presented for the number of subsets required to reach the desired tolerance $\xi_f = 80$, amount of model evaluations, and metric evaluation. As expected, the higher the K, the larger the amount of model evaluations needed; however, the results also show that this parameter has negligible influence on the mean value and spread of the metric ρ and on the amount of required subsets.

Hence, a natural research question arises about whether a minimum K value can be defined in $\mathcal{A}^2$BC-SubSim. To this end, the weighting function ω (recall Equation 3)

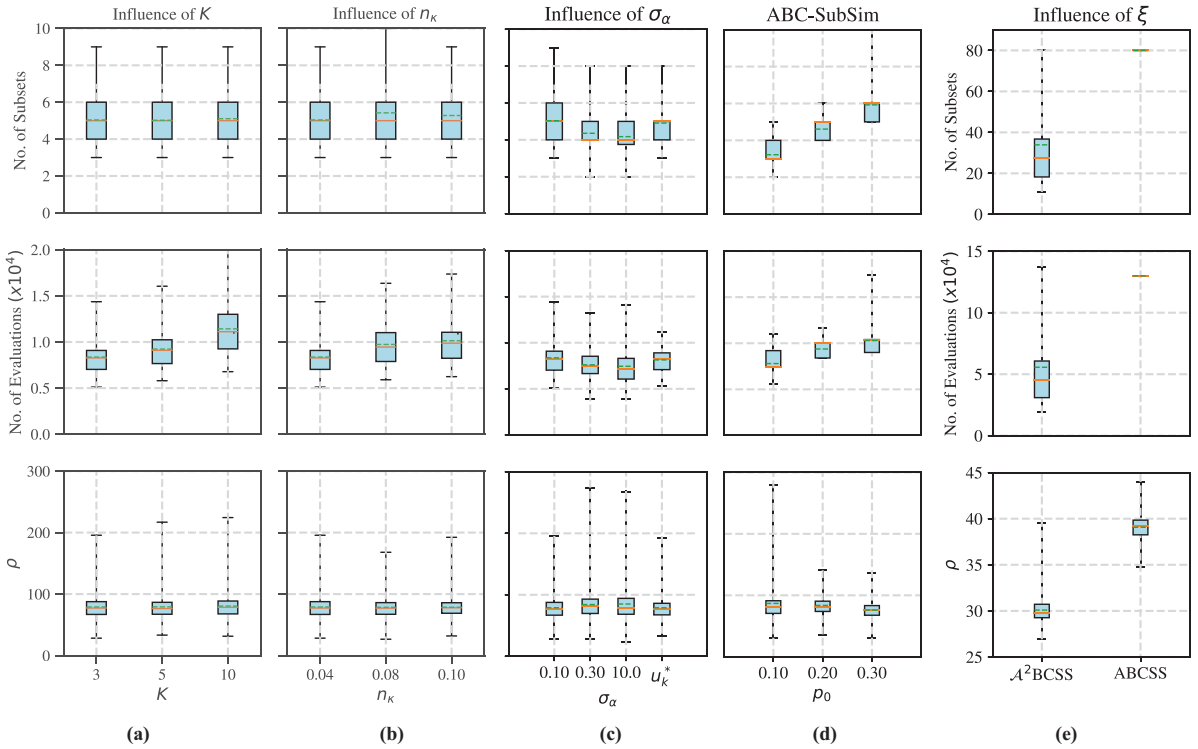


FIGURE 3 Sensitivity study using box-plots of the $\mathcal{A}^2$BC-SubSim hyper-parameters taken from 100 independent runs of the algorithm using $\xi_f = 80.0$ and $N = 2000$ for the cantilever example from Section 2.3. Panel (a) shows the algorithm response under variation of K. In panel (b), the hyper-parameters are fixed to $K = 3$, and $\sigma_\alpha = 0.10$, whereas $n_k = \{0.04, 0.08, 0.10\}$. In panel (c), $n_k = 0.04$, $K = 3$, and $\sigma_\alpha = \{0.1, 0.3, 10\}$. The term u_k^* in panel (c) refers to an alternative piecewise linear function used for comparison (refer to Section 2.2). Panel (d) shows the original ABC-SubSim response using $p_0 = \{0.10, 0.20, 0.30\}$. Panel (e) shows a comparative analysis between ABC-SubSim and $\mathcal{A}^2$BC-SubSim using a more restrictive tolerance $\xi_f = 30$, and $p_0 = 0.2$ for the ABC-SubSim, and $K = 3$, $n_k = 0.04$, and $\sigma_\alpha = 0.1$ for the $\mathcal{A}^2$BC-SubSim

is evaluated for two cases using $K = 3$ and $K = 10$. The results, shown in Figure 4 reveal that there are cases where the $\omega^{(k)}$-values can be fitted using a quadratic downward concave function, whose global maximum lies within any intermediate point of the $p_j^{(k)}$ values (panels a and c); in other cases (panels b and d), the maximum lies above the $p_j^{(k)}$ trials, corresponding to the one with higher ω-value. This suggests that, when using a quadratic downward function, $K = 3$ is enough to obtain the global maximum by interpolation.

Besides, panels (b) and (c) from Figure 3 show summarizing statistics of the $\mathcal{A}^2$BC-SubSim output after the 100 runs by varying n_k and σ_α, respectively, whereas panel (e) shows the comparative performance of ABC-SubSim versus $\mathcal{A}^2$BC-SubSim using a more demanding tolerance, namely $\xi_f = 30$. As with K, the results show that the larger the n_k, the higher the amount of model evaluations required; however, its influence is negligible on the metric distribution and on the amount of subsets. Hence, a proper way to fix n_k is through a sampling formula like the Yamane's formula (Israel, 1992), which give us an

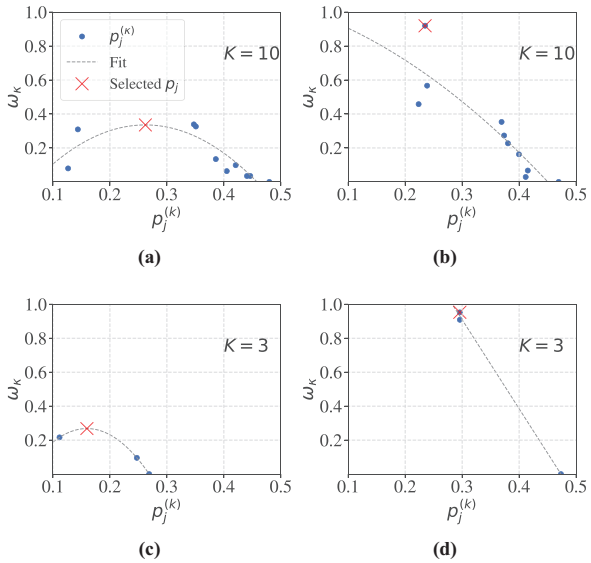


FIGURE 4 Examples of the evaluation of weighting function ω_k for a number of $\{p_j^{(k)}\}_{k=1}^K$ candidates, where $K = 10$ for panels (a,b) and $K = 3$ for panels (c,d)

estimation of a sample size based on an error. Thus, the sample size $n_k N$ can be obtained as follows:

$$n_k N = \frac{N}{1 + NE^2} \tag{9}$$

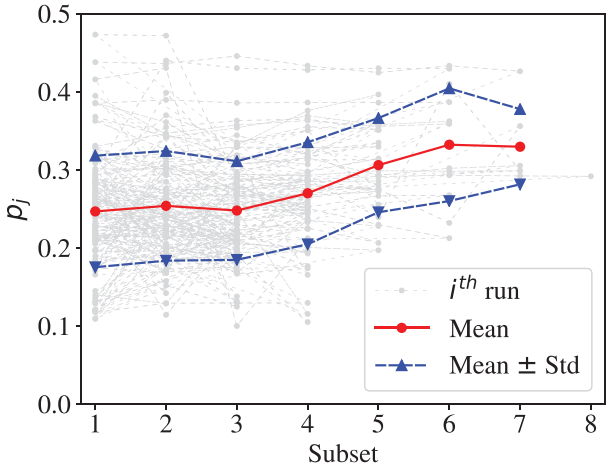
where E is the admissible margin of error, which has been fixed to $E = 0.1$ in this work. Note from the last equation that the adopted $N = 2000$ and $E = 0.1$ lead to an estimated $n_k \simeq 0.04$.

Moreover, the results reveal that the algorithm efficiency is, in general, insensitive to the variation of σ_α, and also that, irrespective of the σ_α adopted, the algorithm behavior is comparable to the ABC-SubSim behavior when $p_0 = 0.1$, near the recommended value, as depicted in panel (d). Additionally, panel (e) shows that $\mathcal{A}^2$BC-SubSim turns to be considerably more efficient than ABC-SubSim when the algorithm is subjected to a more demanding tolerance. In this particular case, the results show that ABC-SubSim was unable to attain the required tolerance $\xi_f = 30$ after 80 simulation levels, whereas $\mathcal{A}^2$BC-SubSim reached it using a mean of 30 simulation levels.

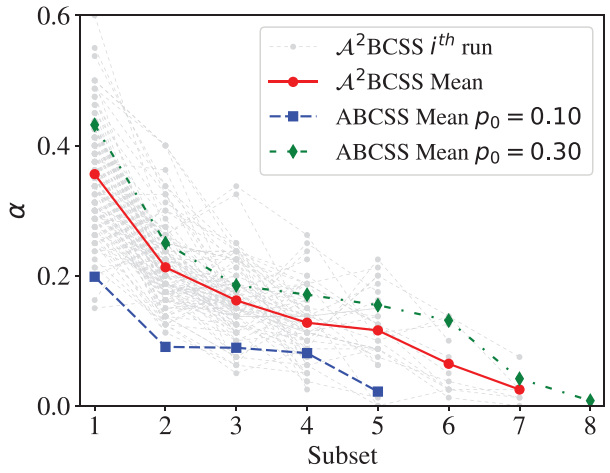
In summary, this analysis shows the small influence of the $\mathcal{A}^2$BC-SubSim hyper-parameters on the algorithm behavior, hence a recommendation is to fix them to $K = 3$, $\sigma_\alpha = 0.1$, whereas n_k can be set using Yamane's formula with an admissible margin of error equal to 10%.

In addition, a comparison in terms of computational cost is carried out between $\mathcal{A}^2$BC-SubSim, using the recommended hyper-parameters (see the first result of either panel a, b, or c in Figure 3) and the original ABC-SubSim (see panel d on the same figure) using $p_0 = 0.2$, as recommended in Chiachio et al. (2014). The results show that, in average, the former required 12% less evaluations than the latter, depicting an improved computational efficiency. Using an Intel®Core™ i7-7700K CPU, 4.20 GHz processor, with 64 GB of RAM running on Windows 10-64 bits, on Python 3.8.3, Spyder 4.1.4, the ABC-SubSim required 504 s for the metric evaluations of 100 runs, while $\mathcal{A}^2$BC-SubSim required 441 s. Also, the $\mathcal{A}^2$BC-SubSim got better measure of the accuracy in the posterior distribution, as the metric function reached 17% lower values. These results, which demonstrate some improvement of both accuracy and efficiency, are in addition to the significant computational savings obtained by avoiding the manual scaling of the hyper-parameter p_0.

As final exercise, the sequence of p_j has been obtained to show how these values evolve as the algorithm progresses. The results are presented in Figure 5a after 100 independent runs of the $\mathcal{A}^2$BC-SubSim algorithm. Note that, in average, the p_j sequence follows a random path with a marked tendency to lie within the range of $p_j = [0.2, 0.3]$. Figure 5b shows the corresponding acceptance rate (α) per simulation level. Displayed in Figure 5b are also shown the mean acceptance rate values obtained from ABC-SubSim algorithm after 100 independent runs using $p_0 = 0.1$ (in blue squares dotted line) and $p_0 = 0.3$ (green rhomboids dotted line). Observe that the mean acceptance rate values of ABC-SubSim constitute average bounds of the $\mathcal{A}^2$BC-SubSim mean acceptance rate.



(a) Sequence of p_j values per simulation level



(b) Acceptance rate (α) per simulation level

FIGURE 5 Results from 100 independent runs using $\mathcal{A}^2$BC-SubSim, with $K = 3$, $n_\kappa = 0.04$ and $\sigma_\alpha = 0.10$ for the column example from Section 2.3. Also shown the equivalent mean results using ABC-SubSim with constant $p_0 = 0.10$ and $p_0 = 0.30$

3 | CASE STUDY: CANTILEVER REINFORCED CONCRETE BEAM-COLUMN NON-LINEAR DEGRADING MODEL

In this section, the $\mathcal{A}^2$BC-SubSim algorithm is used to infer damage parameters from a nonlinear mechanical model of

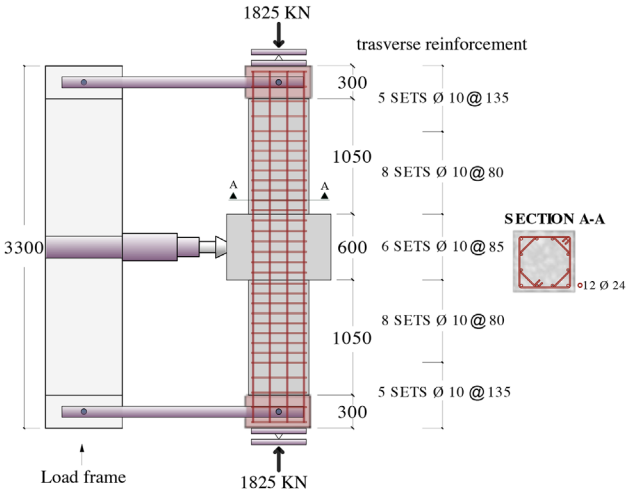


FIGURE 6 General geometry, reinforcement details, and test setup, adapted from Gill (1979). Length units are expressed in millimeters

TABLE 1 Input parameter values of *Concrete01* constitutive model taken for the engineering case study of Section 3

Concrete	f'_c (MPa)	f_{cu} (MPa)	ϵ_{c0} (%)	ϵ^*_{cu} (%)
Confined	−34.70	−23.60	−0.641	−7.110
Unconfined	−23.10	−12.00	−0.183	−0.582

Note: f'_c, concrete peak stress; f_{cu}, concrete ultimate stress; ϵ_{c0}, concrete strain at peak stress; ϵ_{cu}, concrete ultimate strain.

a reinforced concrete column subjected to a constant axial load and cyclic lateral deformation. The column is 3300 mm high and 550 × 550 mm cross-section, with longitudinal reinforcement ratio of 0.019 and 50 mm of coating, as depicted in Figure 6. The average compressive strength of the concrete is 23.1 MPa, whereas the yield strength of longitudinal steel is 375 MPa. The transverse reinforcement is made of two 10-mm diameter stirrups with 297 MPa of yield strength and arranged as shown in Figure 6.

3.1 | Nonlinear forward model

The nonlinear model consists of a force-based beam-column type element in cantilever along with a rotational spring modeled as a zero-length finite element, as depicted in Figure 7. The numerical implementation is carried out using the *OpenSeespy* software (Zhu et al., 2018). Thus, the reinforced concrete section is modeled by an OpenSees *fiber* section as shown in Figure 7 using the *Concrete01* material, whose input values are given in Table 1 for the cases of confined and unconfined concrete. These values have been set following the recommendations proposed by Karthik and Mander (2011) and the estimation of the confinement ratio proposed by Mander et al. (1988). The steel

TABLE 2 Nominal parameter values of hysteretic constitutive model taken for the engineering case study of Section 3

Steel	F_y (MPa)	F_{sh} (MPa)	ϵ_y (%)	ϵ_{sh} (%)
Reinforcement	375.00	468.80	0.188	1.5

fibers for the longitudinal steel reinforcement are modeled using the OpenSees *Hysteretic* material and the recommended properties by the ASCE/COPRI (2014) regulations.

Three damage types are adopted to model the structural deterioration, namely: (1) damage due to ductility, (2) damage due to dissipated energy, and (3) unloading stiffness degradation (β), where parameters D_1, D_2, and β, are involved, respectively, as shown in Figure 8. The nonlinear model is parameterized with a set of model parameters $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_8\}$, where θ_1 to θ_4 act as modifying factors of the physical parameters $\{F_y, F_{sh}, \epsilon_y, \epsilon_{sh}\}$, while θ_5 to θ_8 represent the parameters $\{p_x, p_y, D_1, D_2\}$, respectively. The parameter β was also set equal to θ_8. The reader is referred to Figures 7 and 8 for a schematic description of these parameters within the context of the constitutive equations of the hysteretic material whereas the nominal values adopted for the physical parameters are provided in Table 2. Finally, for the case of the rotational spring element, the *Bond SP01* OpenSees material has been adopted using the recommendations by Zhao and Sritharan (2007) to model the steel, which allows us to take into consideration the effect of strain penetration happening in the anchorage length of the steel reinforcement. Also, the recommendations by Coleman and Spacone (2001) have been adopted to define the ultimate strain corresponding to the constitutive behavior of concrete fibers, as a function of the size of integration points of the frame beam-column elements (i.e., regularization). In this work, Newton–Cotes integration method with five integration points is used. Indeed, the ϵ^*_{cu} values shown in Table 1 are obtained after the aforementioned regularization procedure.

Then, through inference of the referred model parameters, the nonlinear mechanical model can be updated based on experimental data.

3.2 | Results

For the inference of parameters $\boldsymbol{\theta}$, force–displacement pairs taken from the column shown in Figure 6 are considered as data $\mathcal{D}$. The values are taken specifically from the specimen 1 in Berry et al. (2004) open-access database. Such specimen was axially loaded with 1815 KN and also subjected to a lateral cyclic displacement varying from ±5 to ±35 mm. Further information about the test and the data can be found in Berry et al. (2004). The $\mathcal{A}^2$BC-SubSim

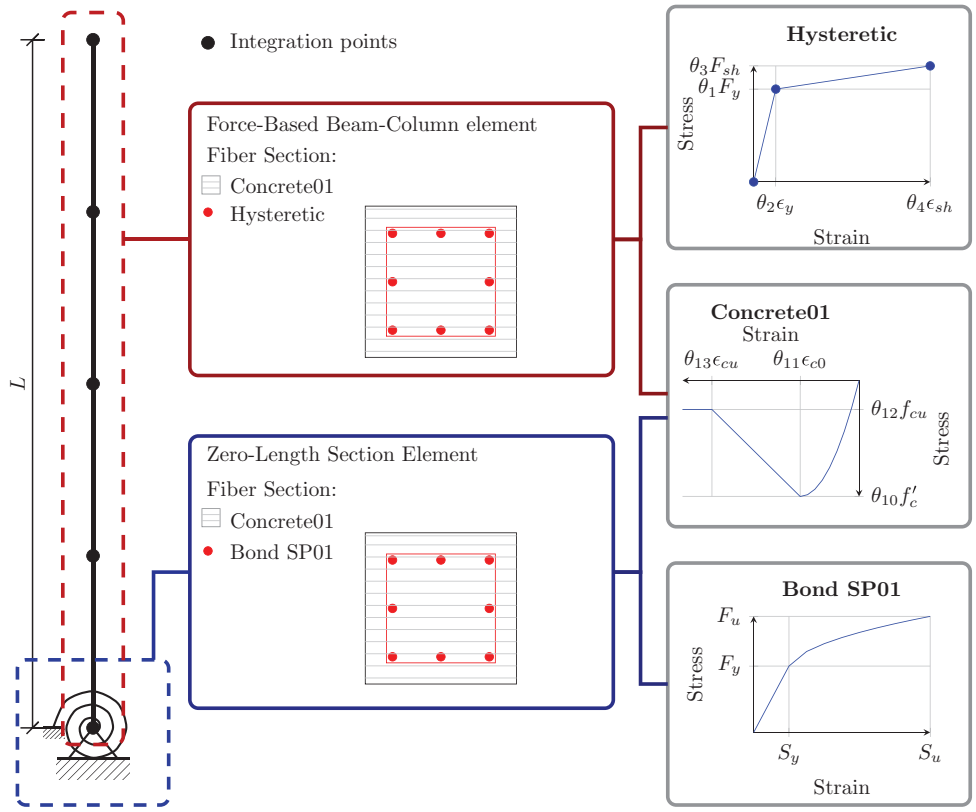


FIGURE 7 Schematic view of the proposed nonlinear model of a reinforced concrete beam-column modeled using OpenSees. On the right-hand side, plots of the constitutive material monotonic behavior are presented, which include information about model parameterization

TABLE 3 Hyper-parameter values adopted for the engineering case study of Section 3

Parameter	K	n_k	σ_α	N	ξ_f	m
Value	3	0.05	0.10	2000	27,550 [N]	30

TABLE 4 Interval definition of the θ parameter space for the case study of Section 3. (shown values are dimensionless)

Parameter	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8
Lower bound	0.50	0.50	0.50	0.50	0.00	0.50	0.00	0.00
Upper bound	1.50	1.50	1.50	1.50	0.50	1.00	0.25	0.25

algorithm is applied by using the configuration given in Table 3 and adopting a $\mathbb{L}_1$ norm as metric function over the force-displacement pairs, that is,

$$\rho_{(\mathbf{x},\theta,D)} = \sum_{i=1}^{n_d} \|x_i - y_i\| \tag{10}$$

where y_i is the ith lateral force data value from the test results, and x_i is the OpenSees ith force value response obtained according to model parameters θ. Uniform PDFs taken over the intervals shown in Table 4 are adopted as a prior distribution of the model parameters θ .

The $\mathcal{A}^2$BC-SubSim results are presented in Figure 9 in terms of approximate posterior PDF $p_\xi(\theta|D)$, while Table 5 provides some summarizing statistics about the aforementioned PDFs. Note from Figure 9 that the approximate posterior of θ_4 and θ_8 are less informative than their component-wise counterparts, meaning that they capture less information from the data, thus their influence within the model to reproduce the data are comparatively lower. Moreover, Figure 11a depicts the hysteretic response predicted by the model using the MAP of the model parameters θ obtained by $\mathcal{A}^2$BC-SubSim versus the test curve showing that the model inferred using $\mathcal{A}^2$BC-SubSim algorithm can satisfactory reproduce the hysteric response, and that this model response is more accurate than the one using the current practice of a time-consuming and costly by-hand calibration method.

4 | DISCUSSION

4.1 | Comparison with current practice trial and error model calibration

As a first step to compare the potential of the proposed methodology, a comparative analysis is carried out by

TABLE 5 Posterior 5th and 95th percentiles, mean, standard deviation, and θ_{MAP} values of θ

Parameter	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8
5th θ	0.715	1.212	0.906	0.529	0.306	0.511	0.002	0.008
95th θ	0.968	1.485	1.032	1.473	0.457	0.666	0.020	0.183
Mean (μ)	0.837	1.362	0.964	0.988	0.378	0.580	0.009	0.091
Std. deviation (σ_θ)	0.095	0.091	0.047	0.384	0.057	0.054	0.005	0.064
θ_{MAP}	0.898	1.481	0.992	1.369	0.443	0.668	0.02	0.144

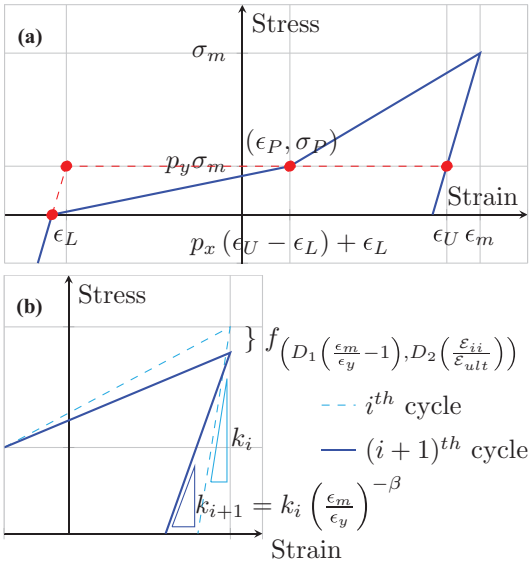


FIGURE 8 (a) Schematic illustration of the in-cycle *Hysteretic* material behavior. (b) Schematic representation of the cyclic degradation (damage) parameters D_1, D_2, and β. The terms ϵ_y, $\mathcal{E}_{ii}$, and $\mathcal{E}_{ult}$, refer to the yield deformation, the ith-cycle absorbed energy, and the maximum available energy, respectively

considering the nonlinear model response calibrated using the MAP of θ from $\mathcal{A}^2$BC-SubSim algorithm, and the one manually calibrated following Barreiro (2018), taken as reference. In Barreiro (2018), an ad hoc trial and error methodology enriched with some mechanical constraints was carried out obtaining a reasonable good model calibration using data from Berry et al. (2004). Similar handy calibrations were used by others on different structural models (i.e., Haselton et al., 2008; Sattar & Liel, 2016) with analogous results. As pointed-out in the last section, results shown in Figure 10 reveal a better accuracy of the model response to the data when using the inferred parameters from the $\mathcal{A}^2$BC-SubSim method. To quantify the model prediction improvement, Equation (10) is used to quantify the mismatch between the predicted and test data, rendering an output of 27,357 N of accumulated error for the $\mathcal{A}^2$BC-SubSim inference procedure against 37,265 N for the by-hand method by Barreiro (2018). This difference can be attributed to simplifications required for the manual scaling process leading to cumulative modeling errors. Indeed, in Barreiro (2018), parameters θ_1 to θ_6 were directly fixed from their nominal values and, therefore, they were not considered during the calibration process. Also, both the damage due to energy and the unloading stiffness degradation parameter (recall Figure 8) were constrained to be equal values. The output of the calibration in terms of model parameter values is shown in Table 6 for both methods. Parameters θ_7 to θ_9 in Table 6 refer to damage due to ductility, damage due to energy, and unloading stiffness degradation, respectively (note that the latter two parameters are now differentiated into θ_8 and θ_9, respectively).

The results demonstrate that the proposed inference procedure using the $\mathcal{A}^2$BC-SubSim algorithm has the capability to reduce the human-factor error and reproduce the test data with higher accuracy than the case of using a by-hand calibration. Moreover, these results also show that an indirect inference of the model parameters can be obtained through the proposed method with quantified uncertainty, whereby robust predictions can be obtained. Thus, a richer knowledge of the actual behavior of the constitutive materials is possible through the information provided by the ABC posterior inference, which can be used for structural diagnostics purposes, among others.

4.2 | Model consistency evaluation

The results shown in Sections 3.2 and 4.1 illustrate how the proposed algorithm can efficiently obtain model

TABLE 6 Calibration results from by-hand and $\mathcal{A}^2$BC-SubSim inference procedure. Results shown for the $\mathcal{A}^2$BC-SubSim case correspond to the maximum a posteriori values of the posterior PDFs

Parameter	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9
$\mathcal{A}^2$ BC-SubSim	0.893	1.453	0.992	1.276	0.425	0.657	0.017	0.149	0.149
By-hand	1.00	1.00	1.00	1.00	0.25	0.70	0.006	1.00	0.21

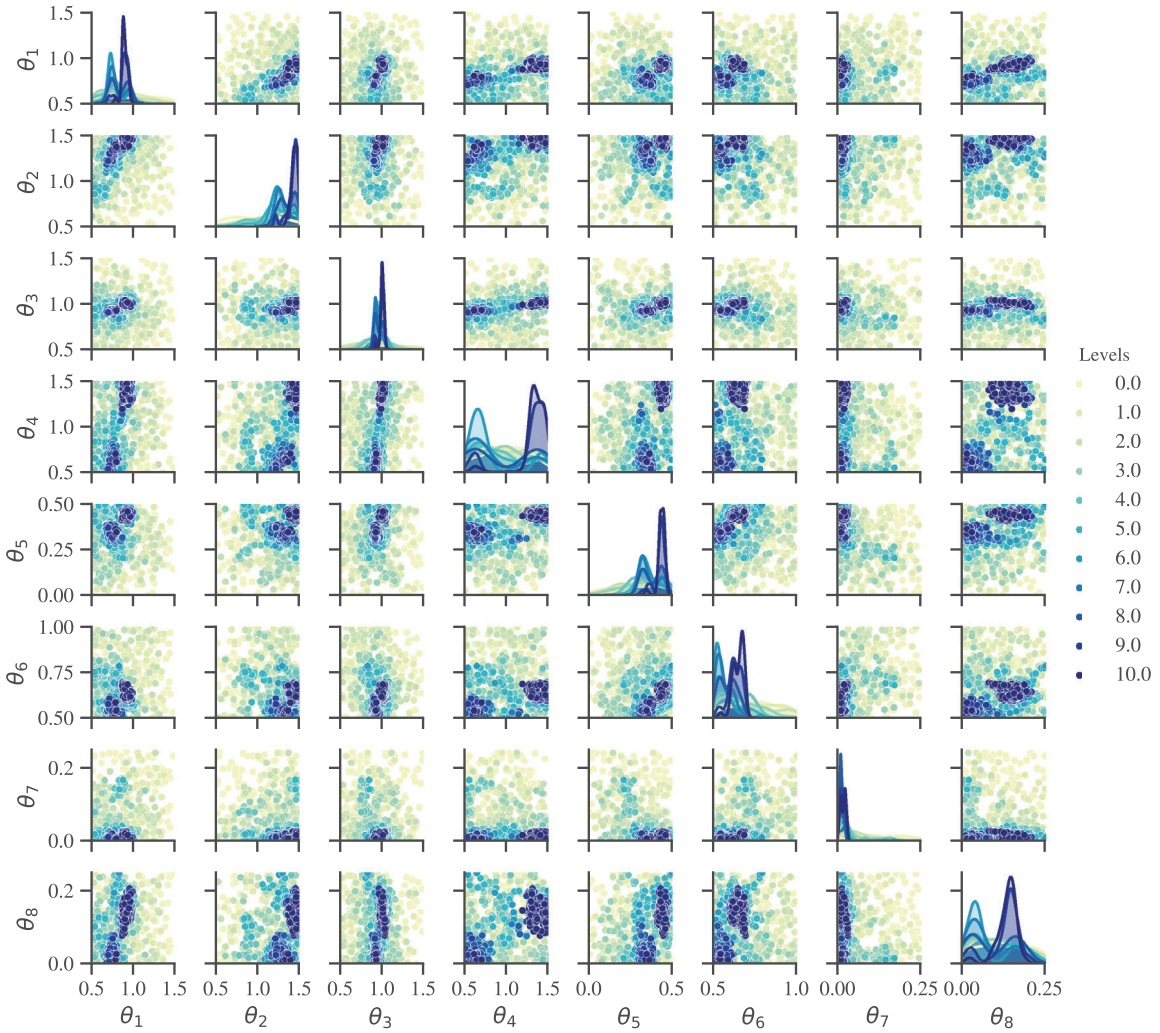
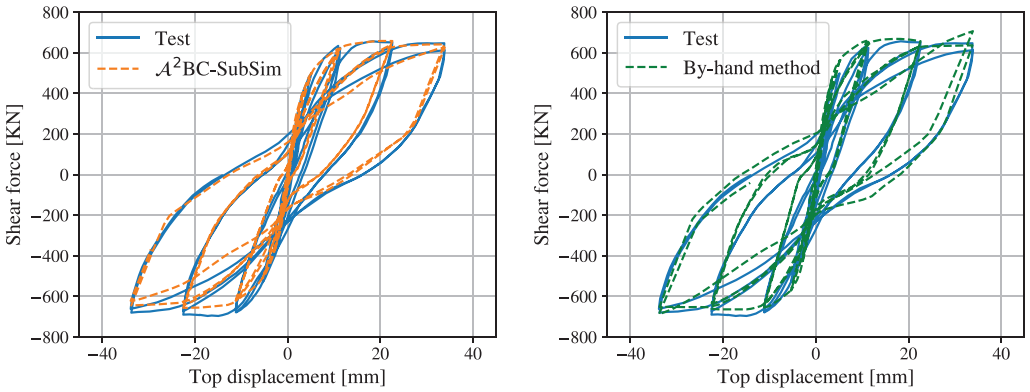
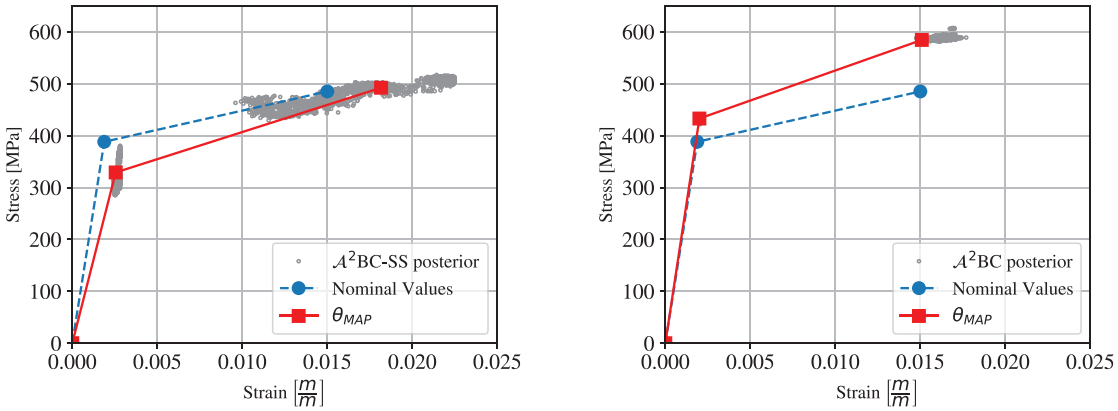


FIGURE 9 Scatter plot representation of the posterior PDF of $\theta = \{\theta_1, \theta_2, \ldots, \theta_8\}$ as $\mathcal{A}^2$BC-SubSim output. On the diagonal, kernel density estimates are shown for the marginal posterior PDFs of the respective parameters



(a) $\mathcal{A}^2$BC-SubSim calibration **(b)** By-hand calibration

FIGURE 10 Results of the comparative analysis of the nonlinear mechanical calibration using (a) $\mathcal{A}^2$BC-SubSim method and (b) the by-hand procedure by Barreiro (2018). The blue line represents the displacement-shear force dataset from Berry et al. (2004)



(a) Inferred steel constitutive model using parameterization given in Section 3.

(b) Inferred steel constitutive model, including concrete behaviour modifiers in the parameterization.

FIGURE 11 Model results after calibration by *ABC* parameter estimation

parameter inference for a nonlinear structural case study using real-world data. However, it is worth mentioning that for this particular application the algorithm itself is unable to give a physically consistent response should the model parameters and their validity range are not properly selected. Thus, engineering judgment is required to validate the results of the model parameter inference. To illustrate this aspect, the steel constitutive model is reproduced after considering the posterior of model parameters $\boldsymbol{\theta}$ inferred using $\mathcal{A}^2$BC-SubSim. Figure 11a shows the inferred MAP value of the elastic module of the steel corresponding to the combined influence of parameters θ_1 and θ_2 (in red) that results about 40% lower than its nominal value (in blue). Note that the latter is inconsistent with the physical reality of the steel behavior, and an explanation to this can be given in terms of the algorithm response, which tries to converge to posterior values of the steel in liaison to the low stiffness values of the concrete due to the induced hysteretic damage.

To overcome this misbehavior, the inference is performed again here including the following modification of model parameterization: (i) A new parameter θ_9 is used to replace θ_8 for the β parameter shown in Figure 8, using $p(\theta_9) = \mathcal{U}(0.00, 0.25)$ as prior PDF; (ii) Parameters θ_{10} and θ_{12} are introduced as modification factors for peak and ultimate stress values of the concrete constitutive model, as shown in Figure 7. The *prior* distributions adopted for these new parameters are taken as $p(\theta_{10}) = p(\theta_{12}) = \mathcal{U}(0.50, 1.50)$; (iii) Analogously, parameters θ_{11} and θ_{13} are used as modification factors for the strain values corresponding to peak and ultimate strain values of the concrete constitutive model, as shown in Figure 7. The prior distributions adopted for these parameters are assumed as $p(\theta_{11}) = p(\theta_{13}) = \mathcal{U}(0.50, 1.50)$. The $\mathcal{A}^2$BC-SubSim algorithm configuration taken for this study is the same as the one shown in Section 3.2 (recall Table 3) except for the following changes: $n_p = 13$, $\xi_f = 20{,}000$ N and $m = 15$. Note that, in this case, ξ_f is set to a lower value with respect to the one chosen in Section 3.2, as the new parameterization allows reducing it to $\xi_f = 20{,}000$ N with similar computational cost. Results are displayed in Table 7. Note that in this case, the inferred elastic module of the steel reinforcement is consistent with the expected steel behavior, as shown in Figure 11b. Also, the new results are consistent with the typical dispersion expected in material stiffness, particularly for the 45% reduction of the concrete stiffness from its nominal value, which is an acceptable value. Also, a closer approximation was achieved in terms of the mismatch function, with an error equal to 23,724 N (i.e., $\xi_f \approx 11.3\%$). In summary, this discussion reveals that the solely use of $\mathcal{A}^2$BC-SubSim algorithm, as with many other inference algorithms, is not enough for an effective material diagnostics and further model calibration of complex nonlinear models, and that engineering judgment is required to make the algorithm outputs consistent with the physics of the problem.

4.3 | Calibration of group of structural tests

The previous results have demonstrated the effectiveness of the $\mathcal{A}^2$BC-SubSim inference method to be used for structural nonlinear model calibration using data from one test. However, current practice of structural model calibration typically requires the use of a complete database of test results (Haselton et al., 2008). Thus, the previous model calibration by $\mathcal{A}^2$BC-SubSim model parameter inference is extended here for the use of a group of experimental tests.

TABLE 7 Posterior 5th and 95th percentiles, mean, standard deviation, and θ_{MAP} values of θ, including concrete parameters

	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9	θ_{10}	θ_{11}	θ_{12}	θ_{13}
5th θ	1.106	1.052	1.204	1.022	0.481	0.670	0.010	0.205	0.178	0.718	1.327	0.898	0.570
95th θ	1.131	1.093	1.222	1.128	0.499	0.699	0.011	0.240	0.194	0.727	1.350	0.931	0.594
μ	1.119	1.072	1.214	1.086	0.491	0.683	0.010	0.227	0.186	0.723	1.338	0.914	0.581
σ	0.007	0.013	0.006	0.030	0.006	0.008	0.000	0.011	0.005	0.003	0.007	0.011	0.010
θ_{MAP}	1.115	1.066	1.205	1.006	0.497	0.704	0.011	0.209	0.186	0.725	1.340	0.933	0.568

TABLE 8 Algorithm hyper-parameter values adopted for the example in Section 4.3

Parameter	K	n_k	σ_α	N	ξ_f	m
Value	3	0.05	0.10	5000	0.12	18

To this end, the data of four tests were taken from Berry et al. (2004) for the calibration process, all of them belonging to the experimental program published by Gill (1979). The four column specimens considered here have the same length, cross-section and longitudinal steel reinforcement ratio, and strength as detailed in Section 3, but with the following varying aspects: (a) Concrete strength (f'_c); (b) Transverse steel strength, separation, size, and configuration; and (c) the axial load (P).

Clearly, one of the experimental campaign objectives was to study the influence of axial load ratio on the flexural behavior of reinforced concrete beam-column elements. Through this criterion, the $\mathcal{A}^2$BC-SubSim is used to obtain a model calibration using the four specimens taking the axial load ratio as the independent variable to define the degradation parameters. For this purpose, the hyper-parameters of the $\mathcal{A}^2$BC-SubSim algorithm were set as shown in Table 8, whereas the degradation parameters were set as functions of the axial load ratio, as follows:

$$D_1 = \frac{P}{A_g f'_c}\theta_7 + \theta_{14} \tag{11}$$

$$D_2 = \frac{P}{A_g f'_c}\theta_8 + \theta_{15} \tag{12}$$

$$\beta = \frac{P}{A_g f'_c}\theta_9 + \theta_{16} \tag{13}$$

where D_1, D_2 and β are the damage parameters already presented in Figure 8, whereas A_g denotes the concrete gross area of the cross-section. The term $\frac{P}{A_g f'_c}$ represents the axial load ratio. *Prior* PDFs of parameters $\theta_7, \theta_8, \theta_9$ and $\theta_{14}, \theta_{15}, \theta_{16}$ are assumed as $p(\theta_7) = p(\theta_8) = p(\theta_9) = \mathcal{U}(-1.00, 1.00)$, and $p(\theta_{14}) = p(\theta_{15}) = p(\theta_{16}) = \mathcal{U}(0.00, 0.30)$, respectively. The rest model parameters along with their prior PDF definitions are considered the same as in Section 4.2.

For this case, the metric is defined as the complementary value of the product of one minus the normalized difference between the test and model output, which is mathematically described as

$$\rho_{(\mathbf{x},\theta,D)} = 1 - \prod_{j=1}^{4}\left[1 - \frac{\sum_{i=1}^{n_d}\left\|x_i^j - y_i^j\right\|}{\sum_{i=1}^{n_d}\left|y_i^j\right|}\right] \tag{14}$$

where $j = 1, \ldots, 4$ is the test index, and y_i^j and x_i^j are the ith corresponding force value from the jth test and model, respectively.

The results are shown in Figure 12 and reveal a good agreement between the predicted model response using the MAP parameter values and the test results of the group of four columns. The normalized difference between test and predicted model resulted equal to 16.90%, 10.15%, 9.97%, and 11.80% for tests 1–4, respectively, achieving similar tolerance as the calibrations done in previous sections. Finally, as a byproduct of posterior inference of damage parameters, a function of variation of the parameters D_1, D_2, and β has been obtained versus the axial load ratio values, and shown in Figure 13.

4.4 | Calibration of a complex structural model

In this section, the proposed Bayesian methodology is tested in a complex structural application to infer multiple uncertain parameters from a nonlinear dynamical model of a tall building subject to seismic excitation.

The data for model inference have been taken from a recent structural test performed by Pratap and Pujol (2021) over a 17-story physical model subjected to seismic impulse produced by a shake table. In this experiment, every story has same mass with value equal to 250 kg. General information about this test along with the measurement data are available at https://datacenterhub.org/deedsdv/publications/view/564. The specific data considered in this case study have been taken from the *TS1-Run2*

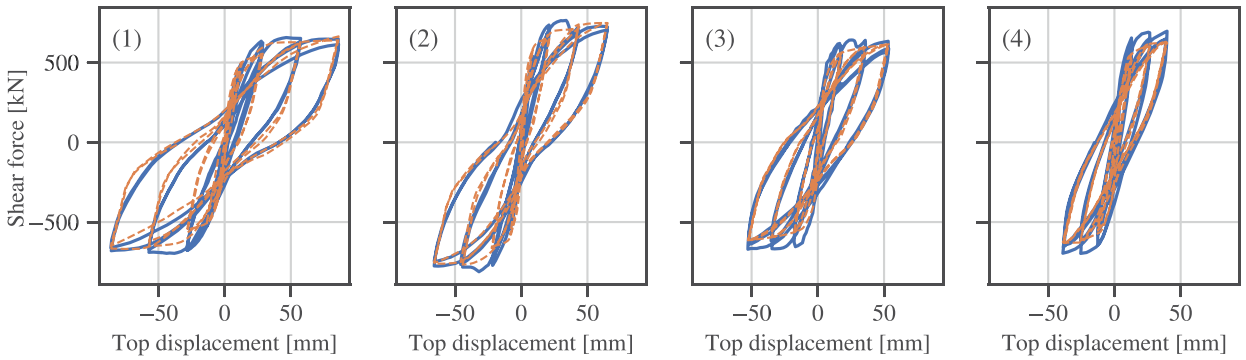


FIGURE 12 Results on four test calibration. Blue lines are the test results and, orange lines, the MAP model prediction

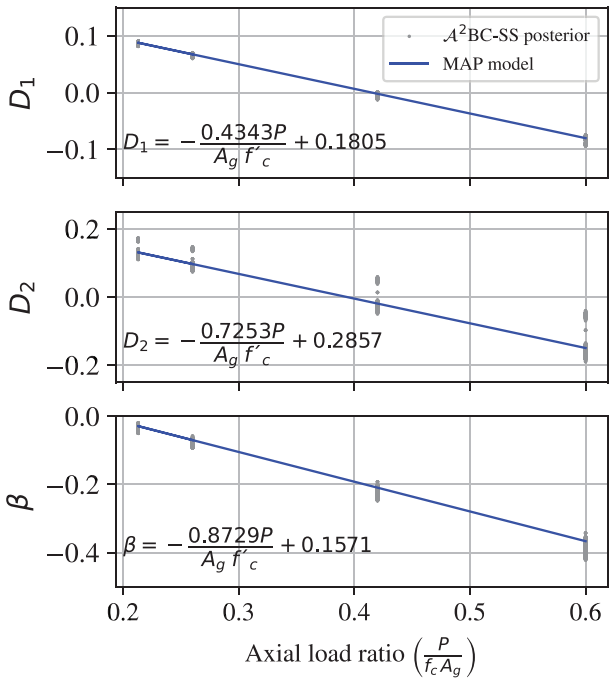


FIGURE 13 Inferred general model of degradation parameters D_1, D_2, and β versus axial load ratio values

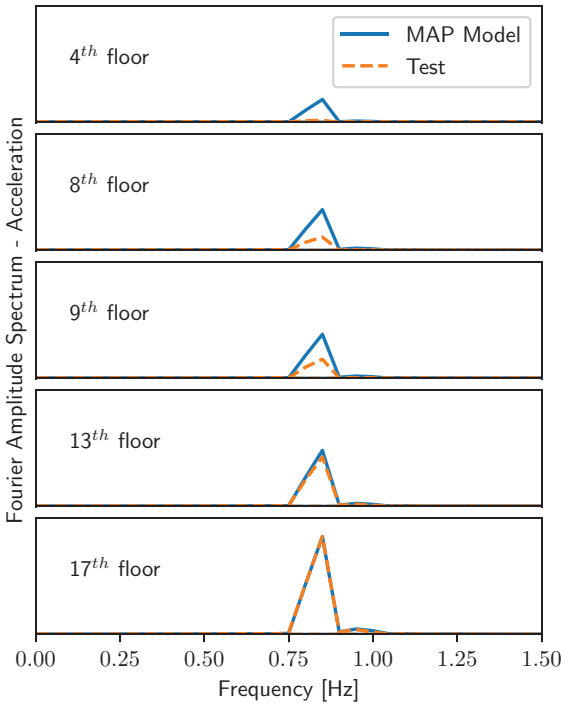


FIGURE 14 Comparison of Fourier amplitude spectrum of the acceleration records of the test measurements and those obtained from the inferred MAP model. The results are presented for stories 4, 8, 9, 13, and 17

experimental run, which contains records of accelerations and displacements of the 4th, 8th, 9th, 13th, and the 17th story. These measurements are presented with a frequency of 1000 Hz, thus preprocessing of the data was required to eliminate noise and reduce the size of the records, which included a band-pass third-order Butterworth filter between the range of frequencies 0.10 and 20 Hz, and a downsampling to a frequency of 100 Hz. Also, a base-line correction was performed to the acceleration records.

The mathematical model is conceived as a 17 in-series mass-spring-dashpots, where the springs are assumed to have an elastic-perfectly plastic behavior. Also, it is assumed that the dynamical response of the building can be fully captured by varying the stiffness, yield strength, and damping values for each floor, which are a priori unknown; therefore, the challenge is to infer them using the proposed Bayesian approach of this paper. More specifically, the parameters considered for this model inference include the elastic stiffness of floor 1–17 (θ_1 to θ_{17}), the yield strength of each floor (θ_{18} to θ_{34}), along with the damping coefficient (θ_{35}) of the dashpots, which is assumed to be equal for every floor. The $\mathcal{A}^2$BC-SubSim algorithm is applied to infer the referred 35 model parameters of the 17-story dynamical model, by adopting the following algorithm

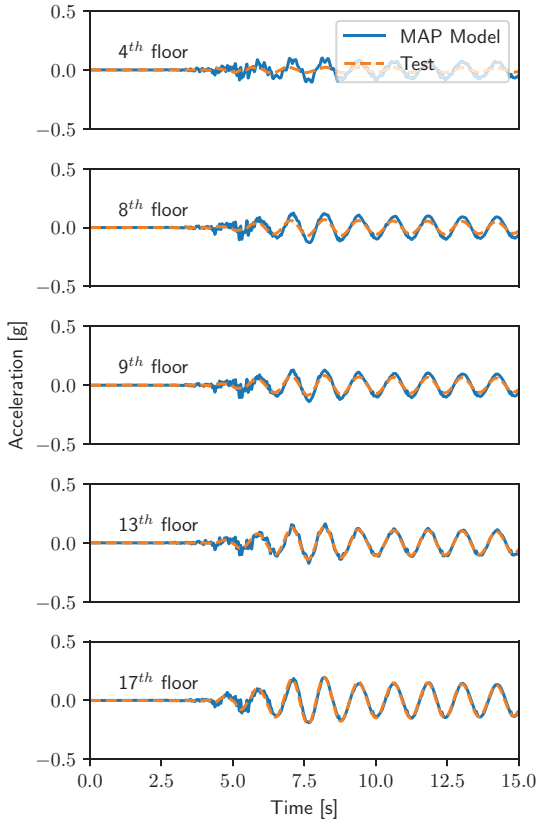


FIGURE 15 Comparison of the acceleration records of the test measurements and those obtained from the inferred MAP model. The results are presented for stories 4, 8, 9, 13, and 17

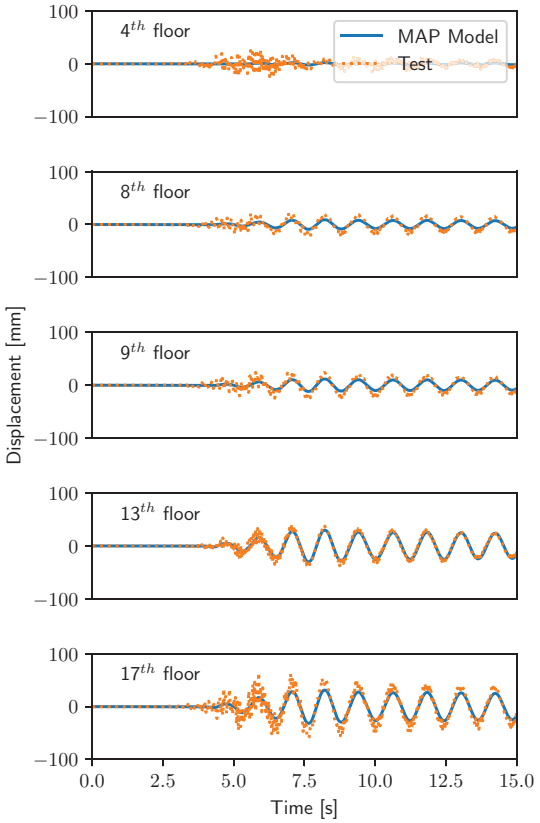


FIGURE 16 Comparison of the displacement records of the test measurements and those obtained from the inferred MAP model. The results are presented for stories 4, 8, 9, 13, and 17

hyper-parameters: $N = 3000$, $\xi_f = 0.04$, $m = 20$, along with the values recommended in Section 2.4 for the rest of hyper-parameters. Uniform PDFs are considered as prior PDFs of the model parameters, such that $p(\theta_1) = p(\theta_2) = \cdots = p(\theta_{17}) = \mathcal{U}[3 \cdot 10^5, 3 \cdot 10^8]$; $p(\theta_{18}) = p(\theta_{19}) = \cdots = p(\theta_{34}) = \mathcal{U}[5 \cdot 10^5, 5 \cdot 10^8]$, and finally $p(\theta_{35}) = \mathcal{U}[1 \cdot 10^3, 1 \cdot 10^5]$, where the units are expressed in SI. In this case, the metric is defined as the relative difference of the Fourier amplitude spectrum of the modeled and measured accelerations, respectively, which is mathematically described as follows:

$$\rho = \sum_i \left| \frac{\hat{\mathbf{a}}^i - \mathbf{a}^i}{\mathbf{A}^i} \right| \tag{15}$$

In the last equation, $\hat{\mathbf{a}}^i$ and $\mathbf{a}^i$ are vectors containing the Fourier amplitude spectrum of the modeled and measured accelerations of the ith story, respectively, whereas $\mathbf{A}^i$ is the resulting sum of the components of the vector $\mathbf{a}^i$. The index $i = \{4, 8, 9, 13, 17\}$.

The results of the inference are shown in Figure 14 that provides a comparison of the Fourier amplitude spectrum of the accelerations obtained from the inferred MAP model, and the respective spectrum using the experimental accelerations. Besides, Figures 15 and 16, respectively, show the accelerations and displacements obtained after the model inference in comparison to their corresponding experimental values. Note that these results are satisfactory as the algorithm has effectively inferred the model parameters so as to reproduce the experimental data with precision, measured by the MAP predictions. Finally, it is worth mentioning that, in average, the $\mathcal{A}^2$BC-SubSim algorithm required $3.5 \cdot 10^3$ s to perform the model inference of the 35 model parameters from this complex structural application, in contrast to the 4.0×10^3 s required by the original ABC-SubSim algorithm (approximately 15% difference). These results, which have been obtained using an Intel® Core™ i7-7700K CPU, 4.20 GHz processor, with 64 GB of RAM running on Windows 10-64 bits, and using Python 3.8.3, demonstrate that the proposed algorithm is able to perform inference in complex nonlinear models with a feasible computational cost. Also, the results confirm again that the computational burden is improved with respect to the original ABC-SubSim algorithm.

5 | CONCLUDING REMARKS

This paper presented a new ABC algorithm named $\mathcal{A}^2$BC-SubSim. This algorithm is a variant of ABC-SubSim that overcomes the need to tune the hyper-parameter p_0, which highly controls the efficiency of the algorithm. This is carried out by an original adaptive selection of this parameter based on a partial evaluation of the next subset. The examples provided demonstrate both efficiency and efficacy for structural nonlinear model calibration based on test results. A comparison of calibrations done by conventional methods (i.e., "by-hand") demonstrate the advantages of $\mathcal{A}^2$BC-SubSim algorithm for this purpose; however, it is also advised that the specialist's judgment is required to attain consistency in the results. The method is extended to perform the calibration of a set of tests, allowing the inference of degradation parameters that are difficult or impossible to estimate using conventional methods. Finally, the method is applied on a complex structural application to show the capacity to perform the inference within large multi-dimensional nonlinear models using real-world structural health monitoring data. Desirable future work includes:

- Surrogate modeling to perform the parameter inference of complex nonlinear structures.
- Applications on structural damage diagnostics based on health monitoring data.
- Explore new variants for the algorithm to perform Bayesian inference using very large (>100 parameters) or extremely large (>1000 parameters) models.

ACKNOWLEDGMENTS

This work was supported by the SINDE (Research and Development System of the Catholic University of Santiago de Guayaquil, Ecuador) under project Cod. Pres #491/Cod. Int. #170. The first author would also like to thank the University of Granada (Spain) for hosting him during the course of this work. Finally, the authors thank the work of Berry et al. (2004) and Pratap and Pujol (2021) for their valuable set of data.

REFERENCES

Albert, C., Künsch, H. R., & Scheidegger, A. (2015). A simulated annealing approach to approximate Bayes computations. *Statistics and Computing*, *25*(6), 1217–1232.

ASCE/COPRI. (2014). *Seismic Design of Piers and Wharves* (ASCE/COPRI 61-14 ed.). American Society of Civil Engineers. https://ascelibrary.org/doi/abs/10.1061/9780784413487

Au, S., & Beck, J. (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, *16*(4), 263–277.

Baragatti, M., Grimaud, A., & Pommeret, D. (2013). Likelihood-free parallel tempering. *Statistics and Computing*, *23*(4), 535–549.

Barreiro, J. (2018). Calibración de un modelo de elementos finitos de hormigón armado no-lineal para incluir los efectos de degradación por carga cíclica parte 4. Universidad Católica de Santiago de Guayaquil.

Beaumont, M. A., Cornuet, J.-M., Marin, J.-M., & Robert, C. P. (2009). Adaptive approximate bayesian computation. *Biometrika*, *96*(4), 983–990.

Beck, J. L. (2010). Bayesian system identification based on probability logic. *Structural Control and Health Monitoring*, *17*(7), 825–847.

Ben Abdessalem, A., Dervilis, N., Wagg, D., & Worden, K. (2019). Model selection and parameter estimation of dynamical systems using a novel variant of approximate bayesian computation. *Mechanical Systems and Signal Processing*, *122*, 364–386.

Berry, M., Parrish, M. & Eberhard, M. (2004). PEER Structural Performance Database User' s Manual (Version 1.0). Number January. PEER.

Betz, W. (2017). *Bayesian inference of engineering models* (Doctoral dissertation). Technische Universitat Munchen.

Bharti, A., & Pedersen, T. (2019). Calibration of stochastic channel models using approximate Bayesian computation. In Douglas Zuckerman (Ed.), *IEEE GLOBECOM Workshops*. IEEE.

Bianconi, F., Tomassoni, L., Antonini, C., & Valigi, P. (2019). A new Bayesian methodology for nonlinear model calibration in Computational Systems Biology. *Frontiers in Applied Mathematics and Statistics*. https://doi.org/10.3389/fams.2020.00025

Chiachio, M., Beck, J. L., Chiachio, J., & Rus, G. (2014). Approximate Bayesian computation by subset simulation. *SIAM Journal on Scientific Computing*, *36*(3), A1339–A1358.

Ching, J., Muto, M., & Beck, J. L. (2006). Structural model updating and health monitoring with incomplete modal data using Gibbs sampler. *Computer-Aided Civil and Infrastructure Engineering*, *21*(4), 242–257.

Christopher, J. D., Wimer, N. T., Lapointe, C., Hayden, T. R. S., Grooms, I., Rieker, G. B., & Hamlington, P. E. (2018). Parameter estimation for complex thermal-fluid flows using approximate Bayesian computation. *Physical Review Fluids*, *3*(10), 104602.

Coleman, J., & Spacone, E. (2001). Localization issues in force-based frame elements. *Journal of Structural Engineering*, *127*(11), 1257–1265.

Da Costa, J. M. J., Orlande, H. R. B., & da Silva, W. B. (2018). Model selection and parameter estimation in tumor growth models using approximate Bayesian computation-ABC. *Computational and Applied Mathematics*, *37*(3), 2795–2815.

De, S., Brewick, P. T., Johnson, E. A., & Wojtkiewicz, S. F. (2019). A hybrid probabilistic framework for model validation with application to structural dynamics modeling. *Mechanical Systems and Signal Processing*, *121*, 961–980.

Del Moral, P., Doucet, A., & Jasra, A. (2012). An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and computing*, *22*(5), 1009–1020.

Dutta, R., Brotzakis, Z. F., & Mira, A. (2018). Bayesian calibration of force-fields from experimental data: TIP4P water. *Journal of Chemical Physics*, *149*(15), 1–26.

Dutta, R., Schoengens, M., Onnela, J.-P. & Mira, A. (2017). ABCpy: A User-Friendly, Extensible, and Parallel Library for Approximate Bayesian Computation. *Proceedings of the Platform for Advanced Scientific Computing Conference (PASC 17)* (vol. 8 pp. 1–9). Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3093172.3093233

Gill, W. (1979). *Ductility of rectangular Reinforced Concrete Columns with axial load* (Doctoral dissertation), University of Canterbury.

Haselton, C. B., Liel, A. B., Lange, S. T., & Deierlein, G. (2008). *Beam-column element model calibrated for predicting flexural response leading to global collapse of RC frame buildings*. (PEER Report 2007).

Israel, G. D. (1992). Determining sample size [Fact sheet]. University of Florida.

Jiang, X., Mahadevan, S., & Adeli, H. (2007). Bayesian wavelet packet denoising for structural system identification. *Structural Control Health Monitoring*, *14*(2), 333–356.

Karabatsos, G., & Leisen, F. (2018). An approximate likelihood perspective on abc methods. *Statistics Surveys*, *12*, 66–104.

Karthik, M. M., & Mander, J. B. (2011). Stress-block parameters for unconfined and confined concrete based on a unified stress-strain model. *Journal of Structural Engineering*, *137*, 270–273.

Kavetski, D., Fenicia, F., Reichert, P., & Albert, C. (2018). Signature-domain calibration of hydrological models using approximate Bayesian computation: Theory and comparison to existing applications. *Water Resources Research*, *54*(6), 4059–4083. https://doi.org/10.1002/2017WR020528

Kulakova, L. (2017). *Bayesian uncertainty quantification for data-driven applications in engineering and life sciences* (Doctoral dissertation). ETH Zurich.

Lam, H. F., Yang, J. H., & Au, S. K. (2018). Markov chain Monte Carlo-based Bayesian method for structural model updating and damage detection. *Structural Control and Health Monitoring*, *25*(4), 1–22.

LeBorgne, M., & Ghannoum, W. (2014). Calibrated analytical element for lateral-strength degradation of reinforced concrete columns. *Engineering Structures*, *81*, 35–48.

Lee, C. S., & Han, S. W. (2018). Computationally effective and accurate simulation of cyclic behaviour of old reinforced concrete columns. *Engineering Structures*, *173*, 892–907.

Mander J. B., Priestley M. J. N., Park R. (1988). Theoretical Stress-Strain Model for Confined Concrete. *Journal of Structural Engineering*, *114*(8), 1804–1826. https://doi.org/10.1061/(asce)0733-9445(1988)114:8(1804)

Marjoram, P., Molitor, J., Plagnol, V., & Tavare, S. (2003). Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, *100*(26), 15324–15328.

McKinley, T. J., Vernon, I., Andrianakis, I., McCreesh, N., Oakley, J. E., Nsubuga, R. N., Goldstein, M.& White, R. G. (2018). Approximate Bayesian computation and simulation-based inference for complex stochastic epidemic models. *Statistical Science*, *33*(1), 4–18.

Mengersen, K. L., Pudlo, P., & Robert, C. P. (2013). Bayesian computation via empirical likelihood. *Proceedings of the National Academy of Sciences*, *110*(4), 1321–1326.

Neal, P. (2012). Efficient likelihood-free Bayesian computation for household epidemics. *Statistics and Computing*, *22*(6), 1239–1256.

Nishina, K., Hashimoto, S., Imamura, N., Ohashi, S., Komatsu, M., Kaneko, S., & Hayashi, S. (2018). Calibration of forest 137 Cs cycling model "FoRothCs" via approximate Bayesian computation based on 6-year observations from plantation forests in Fukushima. *Journal of Environmental Radioactivity*, *193-194*, 82–90.

Papaioannou, I., Betz, W., Zwirglmaier, K., & Straub, D. (2015). MCMC algorithms for subset simulation. *Probabilistic Engineering Mechanics*, *41*, 89–103.

Park, M., Jitkrittum, W., & Sejdinovic, D. (2016). K2-ABC: Approximate bayesian computation with kernel embeddings. *Proceedings of the Machine Learning Research*, *51*, 398–407.

Perez-Ramirez, C., Amezquita-Sanchez, J., Valtierra-Rodriguez, M., Adeli, H., Dominguez-Gonzalez, A., & Romero-Troncoso, R. (2019). Recurrent neural network model with Bayesian training and mutual information for response prediction of large buildings. *Engineering Structures*, *178*, 603–615.

Prangle, D. (2016). Lazy abc. *Statistics and Computing*, *26*(1-2), 171–185.

Pratap, P., & Pujol, S. (2021). *Dynamic tests of an idealized long-period structure*. Creative Commons BY-NC-SA 4.0 (Version 1.0). DEEDS.

Rutter, C. M., Ozik, J., DeYoreo, M., & Collier, N. (2019). Microsimulation model calibration using incremental mixture approximate Bayesian computation. *Annals of Applied Statistics*, *13*(4), 2189–2212.

Sala, M., & Soriguera, F. (2020). Lane-changing and freeway capacity: A Bayesian inference stochastic model. *Computer-Aided Civil and Infrastructure Engineering*, *35*(7), 719–733.

Sattar, S., & Liel, A. B. (2016). Seismic performance of nonductile reinforced concrete frames with masonry infill walls—II: Collapse assessment. *Earthquake Spectra*, *32*(2), 819–842. https://doi.org/10.1193/091514EQS141M

Sharma, H., Gardoni, P., & Hurlebaus, S. (2015). Performance-based probabilistic capacity models and fragility estimates for RC columns subject to vehicle collision. *Computer-Aided Civil and Infrastructure Engineering*, *30*(7), 555–569.

Sisson, S. A., Fan, Y., & Beaumont, M. (2018). *Handbook of approximate Bayesian computation*. CRC Press.

Sisson, S. A., Fan, Y., & Tanaka, M. M. (2007). Sequential Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, *104*(6), 1760–1765.

Song, M., Behmanesh, I., Moaveni, B., & Papadimitriou, C. (2019). Modeling error estimation and response prediction of a 10-story building model through a hierarchical Bayesian model updating framework. *Frontiers in Built Environment*, *5*, 1–15.

Stuart, A. M. (2010). Inverse problems: A Bayesian perspective. *Acta numerica*, *19*, 451–559.

Tavaré, S., Balding, D. J., Griffiths, R. C., & Donnelly, P. (1997). Inferring coalescence times from DNA sequence data. *Genetics*, *145*(2), 505–518.

Tiboaca, O.-D. (2016). *On the Application of the reversible jump Markov chain Monte Carlo method within structural dynamics by Oana-Daniela Tiboaca* (Doctoral dissertation). University of Sheffield.

Toni, T., Welch, D., Strelkowa, N., Ipsen, A., & Stumpf, M. P. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, *6*(31), 187–202.

Yin, T., & Zhu, H.-P. (2020). An efficient algorithm for architecture design of Bayesian neural network in structural model updating. *Computer-Aided Civil and Infrastructure Engineering*, *35*(4), 354–372.

Yuen, K.-V., Kuok, S.-C., & Dong, L. (2019). Self-calibrating bayesian real-time system identification. *Computer-Aided Civil and Infrastructure Engineering*, *34*(9), 806–821.

Zhang, J., Wan, C., & Sato, T. (2013). Advanced Markov chain Monte Carlo approach for finite element calibration under uncertainty. *Computer-Aided Civil and Infrastructure Engineering*, *28*(7), 522–530.

Zhao, J., & Sritharan, S. (2007). Modeling of strain penetration effects in fiber-based analysis of reinforced concrete structures. *ACI Structural Journal*, *104*(2), 133.

Zhu, M., McKenna, F., & Scott, M. H. (2018). Openseespy: Python library for the opensees finite element framework. *SoftwareX*, *7*, 6–11.

Zhu, W., Marin, J. M., & Leisen, F. (2016). A bootstrap likelihood approach to Bayesian computation. *Australian & New Zealand Journal of Statistics*, *58*(2), 227–244.

How to cite this article: Barros J, Chiachío M, Chiachío J, Cabanilla F. Adaptive approximate Bayesian computation by subset simulation for structural model calibration. *Comput Aided Civ Inf*. 2021;1–20. https://doi.org/10.1111/mice.12762