ESAIM: M2AN 48 (2014) 207–229
DOI: 10.1051/m2an/2013097

**ESAIM: Mathematical Modelling and Numerical Analysis**
www.esaim-m2an.org

# ACCURATE AND ONLINE-EFFICIENT EVALUATION OF THE *A POSTERIORI* ERROR BOUND IN THE REDUCED BASIS METHOD

Fabien Casenave[1], Alexandre Ern[1] and Tony Lelièvre[1,2]

**Abstract.** The reduced basis method is a model reduction technique yielding substantial savings of computational time when a solution to a parametrized equation has to be computed for many values of the parameter. Certification of the approximation is possible by means of an *a posteriori* error bound. Under appropriate assumptions, this error bound is computed with an algorithm of complexity independent of the size of the full problem. In practice, the evaluation of the error bound can become very sensitive to round-off errors. We propose herein an explanation of this fact. A first remedy has been proposed in [F. Casenave, Accurate *a posteriori* error evaluation in the reduced basis method. *C. R. Math. Acad. Sci. Paris* **350** (2012) 539–542.]. Herein, we improve this remedy by proposing a new approximation of the error bound using the empirical interpolation method (EIM). This method achieves higher levels of accuracy and requires potentially less precomputations than the usual formula. A version of the EIM stabilized with respect to round-off errors is also derived. The method is illustrated on a simple one-dimensional diffusion problem and a three-dimensional acoustic scattering problem solved by a boundary element method.

**Mathematics Subject Classification.** 65N15, 65D05, 68W25, 76Q05.

Received December 5, 2012. Revised June 13, 2013.
Published online January 10, 2014.

## 1. Introduction

In many problems, such as optimization, uncertainty propagation or real-time simulation, one has to evaluate an objective function for a large number of values of some parameters. Evaluating this objective function often implies solving a parametrized partial differential equation for a given parameter value. In an industrial context, one evaluation of the objective function can already be a challenging numerical problem. To keep reasonable computational costs, various model reduction techniques have been developed to speed up computations. We focus on the Reduced Basis (RB) method [29, 36]. This method has been applied to many kinds of problems, including nonlinear problems such as the viscous Burgers equation [40] or the steady incompressible Navier-Stokes equations [39].

*Keywords and phrases.* Reduced basis method, *a posteriori* error bound, round-off errors, boundary element method, empirical interpolation method, acoustics.

[1] Université Paris-Est, CERMICS (ENPC), 6-8 Avenue Blaise Pascal, Cité Descartes, 77455 Marne-la-Vallée, France.

[2] INRIA Rocquencourt, MICMAC Team-Project, Domaine de Voluceau, B.P. 105, 78153 Le Chesnay Cedex, France.

Article published by EDP Sciences

As described in Section 2, the RB method consists in replacing the sequence $\mathcal{P} \ni \mu \overset{E_\mu}{\mapsto} u_\mu \mapsto Q(u_\mu)$ by the sequence $\mathcal{P} \ni \mu \overset{\hat{E}_\mu}{\mapsto} \hat{u}_\mu \mapsto \hat{Q}(\hat{u}_\mu)$. Here, $\mathcal{P}$ denotes the parameter set, $E_\mu : \mu \mapsto u_\mu$ the model problem, $\hat{E}_\mu : \mu \mapsto \hat{u}_\mu$ its lower-dimensional approximation, $Q(u_\mu)$ the quantity of interest, and $\hat{Q}(\hat{u}_\mu)$ its RB approximation. More specifically, the RB method consists in two steps: (i) A so-called offline stage, where solutions to $E_\mu$ for well-chosen values of the parameter $\mu$ are computed. During this stage, $\hat{N}$ problems of size $N$ are solved (with $\hat{N} \ll N$), and some quantities related to the $\hat{N}$ solutions are stored, and (ii) a so-called online stage, where the precomputed quantities are used to solve $\hat{E}_\mu$ for many values of $\mu$. In this stage, a certification of the approximation is possible by means of an *a posteriori* error bound. An important feature in the RB method is the use of an online-efficient error bound. The notion of online-efficiency is defined in Section 2.4. Moreover, the error bound must be as sharp as possible to faithfully represent the error. However, as noticed for example in ([34], pp. 148–149), the error bound is subject to round-off errors, especially for the computation of accurate solutions. This difficulty can be encountered in complex industrial applications in the following two cases. First and most importantly, when the stability constant of the underlying bilinear (or sesquilinear) form is very small, the classical formula for the error bound fails to certify, even at a relatively crude error level, as illustrated in Section 5 where the stability constant is about $10^{-6}$ and the classical error bound stagnates at about $10^{-4}$. Second, in some industrial codes, the single-precision format is used to speed up computations, when high precision is not needed. In this case, the classical formula for the error bound fails to deliver values below $10^{-4}$ for a stability constant of order 1. The purpose of this work is an explanation of these facts and the derivation of a new method to compute the error bound in an accurate and online-efficient way. Additionally, the new formula uses potentially less precomputed quantities than the classical formula.

In Section 2, we briefly recall the main ingredients of the RB method, namely (i) the construction of the reduced problem, (ii) the a posterior error bound, (iii) the notion of online-efficiency, and (iv) the offline stage during which the vectors of the reduced basis are constructed. We then explain in Section 3 why the classical formula for computing the error bound is ill-conditioned in regard of round-off errors. In Section 4, we present our new procedure based on the empirical interpolation method (EIM). A version of the EIM stabilized with respect to round-off errors is also derived, and the various procedures to compute the error bound are compared on a simple one-dimensional diffusion problem. In Section 5, we apply this new procedure to a three-dimensional acoustic scattering problem.

## 2. The reduced basis method

### 2.1. The model problem

We suppose that the problem of interest has the following discrete variational form, depending on a parameter $\mu$ in a parameter set $\mathcal{P}$: for a finite-dimensional space $\mathcal{V}$ of dimension $N$ (with $N \gg 1$ resulting, *e.g.*, from discretization), find $u_\mu \in \mathcal{V}$ such that

$$E_\mu : a_\mu(u_\mu, v) = b(v), \qquad \forall v \in \mathcal{V}, \tag{2.1}$$

where $a_\mu$ is an inf-sup stable bounded sesquilinear form on $\mathcal{V} \times \mathcal{V}$ and $b$ is a continuous linear form on $\mathcal{V}$. We work in complex vector spaces in view of our application to acoustic scattering. In what follows, the complex conjugate of $z \in \mathbb{C}$ is denoted $z^*$. We define the Riesz isomorphism $J$ from $\mathcal{V}'$ to $\mathcal{V}$ such that for all $\forall l \in \mathcal{V}'$ and all $\forall u \in \mathcal{V}$, $(Jl, u)_{\mathcal{V}} = l(u)$, where $(\cdot,\cdot)_{\mathcal{V}}$ denotes the inner product of $\mathcal{V}$ with associated norm $\|\cdot\|_{\mathcal{V}}$. We denote $\beta_\mu := \inf_{u \in \mathcal{V}} \sup_{v \in \mathcal{V}} \frac{|a_\mu(u,v)|}{\|u\|_{\mathcal{V}} \|v\|_{\mathcal{V}}} > 0$ the inf-sup constant of $a_\mu$ and $\tilde{\beta}_\mu$ a computable positive lower bound of $\beta_\mu$. For simplicity, we consider that the linear form $b$ is independent of the parameter $\mu$. The extension to $\mu$-dependent $b$ is straightforward. We refer to the discrete solution $u_\mu$ as the “truth solution”.

## 2.2. The reduced problem

Suppose that a reduced basis, consisting of $\hat{N}$ solutions $u_{\mu_i}$ of $E_{\mu_i}$, $i \in \{1, \ldots, \hat{N}\}$, has already been constructed. To alleviate the notation, we denote $u_i$ the function $u_{\mu_i}$. How the parameters $\mu_i$ are chosen is briefly outlined in Section 2.5. Given a parameter value $\mu \in \mathcal{P}$, the reduced problem is then a Galerkin procedure written on the linear space $\hat{\mathcal{V}} = \mathrm{Span}\{u_1, \ldots, u_{\hat{N}}\} \subset \mathcal{V}$: find $\hat{u}_\mu \in \hat{\mathcal{V}}$ such that

$$\hat{E}_\mu : a_\mu(\hat{u}_\mu, u_j) = b(u_j), \qquad \forall j \in \{1, \ldots, \hat{N}\}. \tag{2.2}$$

The approximate solution on the reduced basis is written as

$$\hat{u}_\mu = \sum_{i=1}^{\hat{N}} \gamma_i(\mu) u_i. \tag{2.3}$$

Recalling the exact and approximate quantities of interest $Q(u_\mu)$ and $\hat{Q}(\hat{u}_\mu)$, respectively, the quality of the approximation for a given $\mu \in \mathcal{P}$ is quantified by the error measure $\|Q(u_\mu) - \hat{Q}(\hat{u}_\mu)\|$. When we obtain a satisfying error measure with $\hat{N} \ll N$, the RB strategy is successful. Two main cases are generally considered: (i) the so-called general-purpose case, where one is interested in the whole solution: $Q = \hat{Q} = \mathrm{Id}$ and $\|\cdot\| = \|\cdot\|_{\mathcal{V}}$, and (ii) the so-called goal-oriented case, where $Q$ is a linear form on $\mathcal{V}$ and $\|\cdot\| = |\cdot|$. The operator $\hat{Q}$ is consistently built so that $\|Q(u_\mu) - \hat{Q}(\hat{u}_\mu)\|$ vanishes for $\mu = \mu_i$, $i \in \{1, \ldots, \hat{N}\}$.

## 2.3. *A posteriori* error bound

In the standard RB method, the *a posteriori* error bound is a residual-based bound. In what follows, we refer to it simply as error bound. Since this error bound is an upper bound, it provides a way to certify the approximation made by the reduced basis.

**Property 2.1** (General-purpose case)**.** *The following error bound holds: For all* $\mu \in \mathcal{P}$,

$$\|u_\mu - \hat{u}_\mu\|_{\mathcal{V}} \leq \mathcal{E}_1(\mu) := \tilde{\beta}_\mu^{-1} \|G_\mu \hat{u}_\mu\|_{\mathcal{V}}, \tag{2.4}$$

*with* $G_\mu$ *the linear map from* $\mathcal{V}$ *to* $\mathcal{V}$ *such that* $\mathcal{V} \ni u \mapsto G_\mu u := J\left(a_\mu(u, \cdot) - b\right) \in \mathcal{V}$.

*Proof.* See [34], Section 4.3.2. □

In the goal-oriented case, one possible approach is to introduce the following dual problem: Find $v_\mu \in \mathcal{V}$ such that

$$E_\mu^d : a_\mu(w, v_\mu) = Q(w), \qquad \forall w \in \mathcal{V}. \tag{2.5}$$

We wrote the dual problem on the same discrete space $\mathcal{V}$, but another space can be considered. A reduced basis procedure is also carried out for the problem $E_\mu^d$, resulting in an approximation $\hat{v}_\mu$ of $v_\mu$. The approximate quantity of interest is then defined as $\hat{Q}(\hat{u}_\mu) := Q(\hat{u}_\mu) - (G_\mu \hat{u}_\mu, \hat{v}_\mu)_{\mathcal{V}}$, where the second term is the so-called dual-based correction.

**Property 2.2** (Goal-oriented case)**.** *The following error bound holds: For all* $\mu \in \mathcal{P}$,

$$\left|Q(u) - \hat{Q}(\hat{u}_\mu)\right| \leq \mathcal{E}_1^{\mathrm{go}}(\mu) := \left(\tilde{\beta}_\mu^d\right)^{-1} \|G_\mu \hat{u}_\mu\|_{\mathcal{V}} \|G_\mu^d \hat{v}_\mu\|_{\mathcal{V}}, \tag{2.6}$$

*where* $G_\mu^d$ *is the linear map from* $\mathcal{V}$ *to* $\mathcal{V}$ *such that* $\mathcal{V} \ni v \mapsto G_\mu^d u := J\left(a_\mu(\cdot, v) - Q\right) \in \mathcal{V}$ *and* $\tilde{\beta}_\mu^d$ *is a computable lower bound of* $\beta_\mu^d = \inf\limits_{u \in \mathcal{V}} \sup\limits_{v \in \mathcal{V}} \dfrac{|a_\mu(v, u)|}{\|u\|_{\mathcal{V}} \|v\|_{\mathcal{V}}}$. *Obviously,* $\beta_\mu^d = \beta_\mu$ *if* $a_\mu$ *is Hermitian.*

*Proof.* See [5], Proposition 23 and [11], Proposition 3.1. □

In what follows, we mainly focus on the general-purpose case. Extensions to the goal-oriented case are straightforward.

## 2.4. Online-efficiency of the RB method

The notion of online-efficiency is central to the RB method.

**Definition 2.3.** The RB method is said to be online-efficient if in the online stage, (i) the reduced problems can be constructed in complexity independent of $N$, and (ii) the error bound can be computed in complexity independent of $N$.

**Definition 2.4.** The sesquilinear form $a_\mu$ is said to depend on $\mu$ in an affine way if there exist $d$ functions $\alpha_k(\mu) : \mathcal{P} \to \mathbb{C}$ and $d$ $\mu$-independent sesquilinear forms $a_k$ bounded on $\mathcal{V} \times \mathcal{V}$ such that

$$a_\mu(u,v) = \sum_{k=1}^{d} \alpha_k(\mu) a_k(u,v), \qquad \forall u, v \in \mathcal{V}. \tag{2.7}$$

In what follows, we always assume that the affine decomposition (2.7) holds. This decomposition is instrumental to achieve online-efficiency.

**Property 2.5.** *If $a_\mu$ depends on $\mu$ in an affine way, then the RB method is online-efficient.*

*Proof.* (i) The reduced matrix writes $(\hat{A}_\mu)_{j,i} = a_\mu(u_i, u_j)$ and the reduced right-hand side $(\hat{B})_j = b(u_j)$, for all $1 \leq i, j \leq \hat{N}$. There holds $\hat{A}_\mu = \sum_{k=1}^{d} \alpha_k(\mu) \hat{A}_k$, where $(\hat{A}_k)_{ij} := a_k(u_i, u_j)$. Therefore, provided the $d$ matrices $\hat{A}_k$ and the vector $\hat{B}$ are precomputed during the offline stage, the reduced problems are constructed in complexity independent of $N$.

(ii) The operator $G_\mu$ inherits the affine dependence of $a_\mu$ on $\mu$ since, for all $u \in \mathcal{V}$,

$$G_\mu u = -Jb + \sum_{k=1}^{d} \alpha_k(\mu) J a_k(u, \cdot) = G_{00} + \sum_{k=1}^{d} \alpha_k(\mu) G_k u, \tag{2.8}$$

where $G_{00} := -Jb \in \mathcal{V}$ and $G_k u := J a_k(u, \cdot) \in \mathcal{V}$ for all $k \in \{1, \ldots, d\}$. Using this affine decomposition and recalling (2.3), we infer

$$\mathcal{E}_1(\mu) = \tilde{\beta}_\mu^{-1} \left\| G_{00} + \sum_{i=1}^{\hat{N}} \sum_{k=1}^{d} \alpha_k(\mu) \gamma_i(\mu) G_k u_i \right\|_{\mathcal{V}}. \tag{2.9}$$

The scalar product on which the norm in (2.9) hinges can be expanded to provide another formula for the error bound (see [34], Eq. (4.61)):

$$\begin{aligned}\mathcal{E}_2(\mu) = \tilde{\beta}_\mu^{-1} \Bigg( &(G_{00}, G_{00})_{\mathcal{V}} + 2\mathrm{Re} \sum_{i=1}^{\hat{N}} \sum_{k=1}^{d} \gamma_i(\mu) \alpha_k(\mu) (G_k u_i, G_{00})_{\mathcal{V}} \\ &+ \sum_{i,j=1}^{\hat{N}} \sum_{k,l=1}^{d} \gamma_i(\mu) \alpha_k(\mu) \gamma_j^*(\mu) \alpha_l^*(\mu) (G_k u_i, G_l u_j)_{\mathcal{V}} \Bigg)^{\frac{1}{2}},\end{aligned} \tag{2.10}$$

which is computed in complexity independent of $N$ in the online stage provided that $(G_{00}, G_{00})_{\mathcal{V}}$, $(G_k u_i, G_{00})_{\mathcal{V}}$ and $(G_k u_i, G_l u_j)_{\mathcal{V}}$ are precomputed during the offline stage, and provided that a lower bound $\tilde{\beta}_\mu$ of the stability constant of $a_\mu$ is also computed in complexity independent of $N$ (which is possible, for example, by the Successive Constraint Method, see [14, 27]). □

An important observation made in [9], and that will be useful below, is that the formula (2.10) defining $\mathcal{E}_2$ can be rewritten in an equivalent way as

$$\mathcal{E}_2(\mu) := \tilde{\beta}_\mu^{-1} \left(\delta^2 + 2\mathrm{Re}(s^t \hat{x}_\mu) + \hat{x}_\mu^{*t} S \hat{x}_\mu\right)^{\frac{1}{2}}, \tag{2.11}$$

where $\delta := \|G_{00}\|_{\mathcal{V}}$, $s$ and $\hat{x}_\mu$ are vectors in $\mathbb{C}^{d\hat{N}}$ with components $s_I := (G_k u_i, G_{00})_{\mathcal{V}}$ and $(\hat{x}_\mu)_I := \alpha_k(\mu)\gamma_i(\mu)$, and $S$ is a matrix in $\mathbb{C}^{d\hat{N},d\hat{N}}$ with coefficients $S_{I,J} := (G_k u_i, G_l u_j)_{\mathcal{V}}$ (with $I$ and $J$ re-indexing respectively $(k,i)$ and $(l,j)$, for all $1 \le k,l \le d$ and all $1 \le i,j \le \hat{N}$). The $t$ superscript denotes the transposition. The vector $s$ and the matrix $S$ depend on the reduced basis functions $\{u_i\}_{1\le i\le \hat{N}}$ but are independent of $\mu$, and the vector $\hat{x}_\mu$ depends on the RB approximation $\hat{u}_\mu$ *via* the coefficients $\gamma_i(\mu)$. Notice that the term between parenthesis on the right-hand side of (2.11) is a multivariate polynomial in $\hat{x}_\mu$ of total degree 2. We would like to stress that $\mathcal{E}_1(\mu) = \mathcal{E}_2(\mu)$ (in infinite precision arithmetic): the indices 1 and 2 are used to denote two different ways to compute the same quantity. In particular, $\mathcal{E}_1(\mu)$ is not online efficient, while $\mathcal{E}_2(\mu)$ is.

### 2.5. The offline stage

Fix a discrete subset of parameters $\mathcal{P}_{\text{trial}} \subset \mathcal{P}$. In the offline stage, the parameters $\mu_i$ (from which the reduced basis is constructed) are chosen by a greedy algorithm as elements of $\mathcal{P}_{\text{trial}}$. We denote $\mathcal{P}_{\text{select}}$ the set of these selected parameters; see [34], Section 3.3 for a presentation of the greedy algorithm. At each step of the algorithm, the new quantities $a_k(u_i, u_j)$ and $b(u_j)$ are computed and stored, as well as the new components of the vector $s$ and of the matrix $S$ to be used in the formula (2.11) for $\mathcal{E}_2$. This task, as that of evaluating $G_{00}$, typically requires inverting the stiffness matrix in $\mathcal{V}$ by solving, for all $k \in \{1,\dots,d\}$ and all $i \in \{1,\dots,\hat{N}\}$, the variational problem: find $w_{i,k} \in \mathcal{V}$ such that

$$E_{G\,i,k} : (w_{i,k}, v)_{\mathcal{V}} = a_k(u_i, v), \qquad \forall v \in \mathcal{V}. \tag{2.12}$$

Then, $G_k u_i = w_{i,k}$ can be computed. The computation of $(G_k u_i, G_l u_j)_{\mathcal{V}}$ follows from the solutions of $E_{G\,i,k}$ and $E_{G\,j,l}$. Since the error bounds are evaluated using the formula $\mathcal{E}_2(\mu)$, for all $\mu \in \mathcal{P}_{\text{trial}}$, with the current state of the reduced basis, finding the maximum of the error bound on $\mathcal{P}_{\text{trial}}$ is of complexity independent of $N$. This allows one to consider very large sets $\mathcal{P}_{\text{trial}}$ without increasing too much the complexity of the whole offline procedure.

## 3. Round-off errors and online certification

In this section, we explain why the online-efficient error bound (2.11) may be sensitive to round-off errors.

### 3.1. Elements of floating-point arithmetic

In a computer, real numbers are represented by a finite number of bits, called floating-point representation. Current architectures are optimized for a format used by a large majority of softwares: IEEE 754 double-precision binary floating-point format. Let $x$ be a real number. The floating point representation of $x$ is denoted by $fl(x)$. When a (nonzero) real number is rounded to the closest floating-point number, the relative error on its floating-point representation is bounded by a number, $\epsilon$, called the machine precision. In double precision, $\epsilon = 5\times 10^{-16}$, see [21], Section 1.2. Let $x$ and $y$ be real numbers. When computing the operation $x+y$, the result returned by the computer can be different from its theoretical value. Whenever the difference is substantial, a loss of significance occurs. A well-known case of loss of significance is when $x$ and $y$ are almost opposite numbers. Suppose that $x = -y$. We denote by $\mathrm{maxfl}(x+y)$ the result that the computer returns when the maximal accumulation of round-off errors occurs when computing the summation. There holds

$$|\mathrm{maxfl}(x+y)| \approx 2\epsilon|x|. \tag{3.1}$$

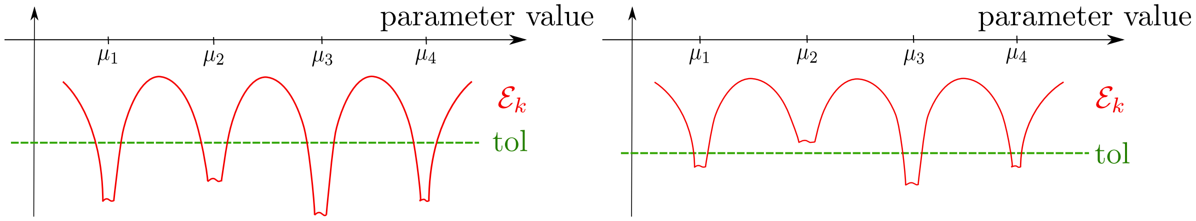

FIGURE 1. Schematic illustration of Definition 3.1, with $\mathcal{P}_{\text{select}} = \{\mu_1, \ldots, \mu_4\}$. Left: the formula $\mathcal{E}_k$ is valid for computing the error bound with tolerance tol; right: the formula is not valid as $\mathcal{E}_k(\mu_2) > \text{tol}$.

When implementing an algorithm, one should ensure that each step is free of such a loss of significance. In some cases, simply changing the order of the operations can prevent these situations. As an illustration, consider $x = 1$, $y = 1 + 10^{-7}$, and the operation $x^2 - 2xy + y^2$. This is a sum of terms where the first intermediate result in the sum is 14 orders larger than the result. Therefore, a loss of significance is expected. The relative error of this computation is about $8 \times 10^{-4}$. Computing $(x - y)^2$, which is the factorization of the considered operation, leads to a relative error of about $10^{-9}$. Thus, the terms of the sum are only 7 orders larger than the results, leading to a less catastrophic loss of significance. In this specific case, the remedy consists in carrying out the sum before the multiplication. In the RB context, the evaluation of the formula $\mathcal{E}_2$ suffers from such a loss of significance, as we now explain.

### 3.2. Validity of the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ for computing the error bound

Consider the two formulae $\mathcal{E}_1$, see (2.9), and $\mathcal{E}_2$, see (2.11), for computing the error bound.

**Definition 3.1.** The formula $\mathcal{E}_k$, $k = 1, 2$, is said to be valid for computing the error bound with tolerance tol if

$$\max_{\mu \in \mathcal{P}_{\text{select}}} (\mathcal{E}_k(\mu)) \leq \text{tol}. \tag{3.2}$$

From a theoretical viewpoint, the error $\|u_\mu - \hat{u}_\mu\|_{\mathcal{V}}$ and the residual $G_\mu u_\mu$ vanish for all $\mu \in \mathcal{P}_{\text{select}}$. Hence, any formula for computing the residual-based error bound vanishes as well and therefore is valid with any tolerance. However, the validity of a formula for computing the error bound is to be considered in the presence of some adverse phenomenon introducing errors in the computation, see Figure 1. The greedy algorithm in the offline stage stops when $\max\limits_{\mu \in \mathcal{P}_{\text{trial}}} (\mathcal{E}_k(\mu)) < \text{tol}_{\text{RB}}$, where $\text{tol}_{\text{RB}}$ denotes the maximum acceptable error made by the RB approximation. Therefore, if the minimum tolerance for which an error bound $\mathcal{E}_k$ is valid is larger than $\text{tol}_{\text{RB}}$, then the greedy algorithm cannot converge and will keep increasing the set $\mathcal{P}_{\text{select}}$ although the error can be actually very small.

We examine the validity of the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ for computing the error bound in the presence of two independent phenomena: round-off errors and approximate reduced basis functions $u_i$ (in the context of inexact linear algebra solvers for $E_{\mu_i}$).

#### *3.2.1. Round-off errors*

We investigate the influence of round-off errors when computing the error bounds $\mathcal{E}_1(\mu)$ and $\mathcal{E}_2(\mu)$. As observed at the end of Section 3.1, the computation of a polynomial using a factorized form is more accurate than using the developed form, in particular at points close to its roots. Here, $(\tilde{\beta}_\mu \mathcal{E}_2(\mu))^2$ is a multivariate polynomial of degree 2 in $\hat{x}_\mu$ computed in a developed form, whereas the scalar product $(G_\mu u_\mu, G_\mu u_\mu)_{\mathcal{V}}$ used in the computation of $\mathcal{E}_1(\mu)$ is not developed.

In this section, we neglect the round-off errors introduced when solving $E_\mu$ and $\hat{E}_\mu$, so that the reduced basis functions $u_i$ and the reduced solutions $\hat{u}_\mu$ are considered free of round-off errors. We also suppose that the computable positive lower bound $\tilde{\beta}_\mu$ of the inf-sup constant is computed free of round-off errors, see Remark 3.4.

**Proposition 3.2.** *Let $\mu \in \mathcal{P}_{\rm select}$ and let* $\operatorname{maxfl}(\tilde{\beta}_\mu \mathcal{E}_{\rm k}(\mu))$, *$k = 1, 2$, denote the evaluation of $\tilde{\beta}_\mu \mathcal{E}_k(\mu)$ when the maximum accumulation of round-off errors occurs. There holds*

$$\begin{aligned} \operatorname{maxfl}\left(\tilde{\beta}_\mu \mathcal{E}_1(\mu)\right) &\geq 2\delta\epsilon, \\ \operatorname{maxfl}\left(\tilde{\beta}_\mu \mathcal{E}_2(\mu)\right) &\geq 2\delta\sqrt{\epsilon}, \end{aligned} \tag{3.3}$$

*where $\delta = \|G_{00}\|_{\mathcal{V}}$ and $\epsilon$ is the machine precision.*

*Proof.* Let $\mu \in \mathcal{P}_{\rm select}$. We present the proof for $\mathcal{E}_1(\mu)$; the proof for $\mathcal{E}_2(\mu)$ is similar. We need to evaluate the right-hand side of (2.9). Let $(\varphi_\rho)_{1\leq\rho\leq N}$ denote the basis of $\mathcal{V}$, so that, for instance, $G_{00} = \sum_{\rho=1}^{N} (G_{00})_\rho \varphi_\rho$. In exact arithmetics, there holds $\mathcal{E}_1(\mu) = 0$, so that $\sum_{i=1}^{\hat{N}} \sum_{k=1}^{d} \gamma_i(\mu)\alpha_k(\mu) (G_k u_i)_\rho = -(G_{00})_\rho$ for all $1 \leq \rho \leq N$. As a result, using (3.1), we obtain

$$\left|\operatorname{maxfl}\left((\mathrm{G}_{00})_\rho + \sum_{\mathrm{i}=1}^{\hat{\mathrm{N}}} \sum_{\mathrm{k}=1}^{\mathrm{d}} \gamma_{\mathrm{i}}(\mu)\alpha_{\mathrm{k}}(\mu)(\mathrm{G}_{\mathrm{k}}\mathrm{u}_{\mathrm{i}})_\rho\right)\right| \approx 2|(G_{00})_\rho|\epsilon.$$

Since computing the $\mathcal{V}$-norm on the right-hand side of (2.9) can only increase the round-off errors, we infer the desired lower bound. □

**Remark 3.3** (Validity of the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$)**.** We indeed observe in our simulations that the round-off errors on $\mathcal{E}_1$ scale like $\epsilon$, while the round-off errors on $\mathcal{E}_2$ scale like $\sqrt{\epsilon}$, see Section 4.3. Then, if we suppose that the lower bounds are reached in (3.3), the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ are valid for computing the error bound with tolerance tol if, respectively,

$$\begin{aligned} &\text{for } \mathcal{E}_1, \qquad 2\left(\tilde{\beta}_{\min}\right)^{-1} \delta\epsilon \leq \text{tol}, \\ &\text{for } \mathcal{E}_2, \qquad 2\left(\tilde{\beta}_{\min}\right)^{-1} \delta\sqrt{\epsilon} \leq \text{tol}, \end{aligned} \tag{3.4}$$

where $\tilde{\beta}_{\min} = \inf_{\mu\in\mathcal{P}_{\rm select}} (\tilde{\beta}_\mu)$.

**Remark 3.4** (Inf-sup constant)**.** The computable positive lower bound $\tilde{\beta}_\mu$ of the inf-sup constant suffers from round-off errors as well. However, since it is a multiplicative factor, the quality of its computation does not severely affect the quality of the error bound. Moreover, the value of the inf-sup constant does not depend on the size of the reduced basis, contrary to $\|G_\mu \hat{u}_\mu\|_{\mathcal{V}}$. Therefore, there is no phenomenon susceptible to degrade the accuracy of its computation with the increase of the size of the reduced basis. If the Successive Constraint Method is used, the procedure to compute $\tilde{\beta}_\mu$ is carried out before the greedy algorithm of the RB method.

**Remark 3.5** (Improved floating-point arithmetic)**.** Increasing the machine precision from $\epsilon$ to $\epsilon^2$ (quadruple-precision) for computing the coefficients in (2.11), as well as for the evaluation of the multivariate polynomial in $\hat{x}_\mu$, is a first solution to recover a good precision with the formula $\mathcal{E}_2$. There are also methods allowing one to double the precision of the evaluation of a polynomial while keeping the double-precision format, namely compensated schemes. For instance, the compensated Horner scheme in double-precision [28] doubles the precision and is faster than the full quadruple precision implementation. However, this corresponds to representing the result of the intermediate operations by two doubles, one for the value in double-precision and another one for the subsequent digits. Therefore, these strategies are equivalent to quadruple precision (except for the computational savings in evaluating the error bound). Moreover, since current architectures are optimized for the double-precision format, changing the floating-point arithmetic can potentially degrade software performance.

**Remark 3.6** (Goal-oriented case, round-off errors)**.** The same analysis can be carried-out in the goal-oriented case. Let $\mu \in \mathcal{P}_{\text{select}}$. There holds

$$\begin{aligned} \text{maxfl}\left(\tilde{\beta}^d_\mu \mathcal{E}^{\text{go}}_1(\mu)\right) &\geq 2\delta\varsigma\epsilon^2, \\ \text{maxfl}\left(\tilde{\beta}^d_\mu \mathcal{E}^{\text{go}}_2(\mu)\right) &\geq 2\delta\varsigma\epsilon, \end{aligned} \tag{3.5}$$

where $\varsigma := \|Q\|_{\mathcal{V}'}$. We indeed observe in our simulations that the round-off errors on $\mathcal{E}^{\text{go}}_1$ scale like $\epsilon^2$, while the round-off errors on $\mathcal{E}^{\text{go}}_2$ scale like $\epsilon$, see Section 5. If we suppose that the lower bounds are reached in (3.5), then the formulae $\mathcal{E}^{\text{go}}_1$ and $\mathcal{E}^{\text{go}}_2$ are valid for computing the error bound with tolerance tol if, respectively,

$$\begin{aligned} &\text{for } \mathcal{E}^{\text{go}}_1, \qquad 2\left(\tilde{\beta}^d_{\min}\right)^{-1}\delta\varsigma\epsilon^2 \leq \text{tol}, \\ &\text{for } \mathcal{E}^{\text{go}}_2, \qquad 2\left(\tilde{\beta}^d_{\min}\right)^{-1}\delta\varsigma\epsilon \leq \text{tol}, \end{aligned} \tag{3.6}$$

where $\tilde{\beta}^d_{\min} = \inf\limits_{\mu \in \mathcal{P}_{\text{select}}} (\tilde{\beta}^d_\mu)$.

*3.2.2. Approximate reduced basis functions*

In large-scale simulations, the accuracy of the RB procedure is also limited by the numerical method used for computing the reduced basis functions. We want here to illustrate this fact on a simple example where we suppose that the approximation of the reduced basis functions comes from an iterative solver with prescribed stopping criterion. We recall that for a given value $\mu \in \mathcal{P}_{\text{select}}$, $E_\mu$ consists in solving a linear system of size $N$ of the form $A_\mu U_\mu = B$. Thus, for $\mu \in \mathcal{P}_{\text{trial}}$, the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ for the error bound are based on the computation of the residual of $E_\mu$ for the reduced solution $\hat{u}_\mu$. Indeed, it is easy to see that $\|G_\mu \hat{u}_\mu\|_{\mathcal{V}} = \|A_\mu \hat{U}_\mu - B\|_{*\mathcal{V}'}$, where for all $\varPhi \in \mathbb{C}^N$, $\|\varPhi\|_{*\mathcal{V}'} = \sup\limits_{V \in \mathbb{C}^N} \frac{\left|(V,\varPhi)_{\mathbb{C}^N}\right|}{\|\sum_{i=1}^N V_i\varphi_i\|_{\mathcal{V}}}$, recalling that $(\varphi_\rho)_{1\leq\rho\leq N}$ are the basis functions in $\mathcal{V}$, see [18], Section 9.1.5.

In this section, we suppose that the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ are free of round-off errors (therefore, for all $\mu \in \mathcal{P}_{\text{trial}}$, $\mathcal{E}_1(\mu) = \mathcal{E}_2(\mu)$), but the problem $E_\mu$ is not solved exactly, leading to approximate reduced basis functions such that the residuals do not vanish. Hence, for all $\mu \in \mathcal{P}_{\text{select}}$, $\mathcal{E}_1(\mu) = \mathcal{E}_2(\mu)$ and these error bounds are nonzero owing to inexact linear algebra solves. The reduced problems $\hat{E}_\mu$ are supposed to be solved freely of round-off errors.

**Proposition 3.7** (Approximate reduced basis functions)**.** *If the reduced basis functions are computed using an iterative solver with the following stopping criterion on the normalized residual:*

$$\forall \mu \in \mathcal{P}_{\text{trial}}, \qquad \frac{\|A_\mu U_\mu - B\|_{*\mathcal{V}'}}{\|B\|_{*\mathcal{V}'}} \leq \xi, \tag{3.7}$$

*then the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ are valid for computing the error bound with tolerance* tol *if*

$$\tilde{\beta}^{-1}_{\min}\delta\xi \leq \text{tol}. \tag{3.8}$$

*Proof.* Let $k \in \{1,2\}$, let $\mu \in \mathcal{P}_{\text{select}}$ and suppose that the stopping criterion (3.7) is satisfied. Then, $\hat{u}_\mu = u_\mu$, but $u_\mu$ does not exactly solve $E_\mu$. First, by definition of the $\|\cdot\|_{*\mathcal{V}}$ norm, $\|B\|_{*\mathcal{V}'} = \sup\limits_{V \in \mathbb{C}^N} \frac{\left|b(\sum_{i=1}^N V_i\varphi_i)\right|}{\|\sum_{i=1}^N V_i\varphi_i\|_{\mathcal{V}}} = \|b\|_{\mathcal{V}'} = \|G_{00}\|_{\mathcal{V}} = \delta$. Then, $\|G_\mu \hat{u}_\mu\|_{\mathcal{V}} = \sup\limits_{v \in \mathcal{V}} \frac{(G_\mu \hat{u}_\mu, v)_{\mathcal{V}}}{\|v\|_{\mathcal{V}}} = \sup\limits_{v \in \mathcal{V}} \frac{a_\mu(\hat{u}_\mu, v) - b(v)}{\|v\|_{\mathcal{V}}} = \sup\limits_{V \in \mathbb{C}^N} \frac{(V, A_\mu \hat{U}_\mu - B)_{\mathbb{C}^N}}{\|\sum_{i=1}^N V_i\varphi_i\|_{\mathcal{V}}} = \|A_\mu \hat{U}_\mu - B\|_{*\mathcal{V}'}$. Therefore,

$$\mathcal{E}_k(\mu) = \tilde{\beta}^{-1}_\mu \|G_\mu \hat{u}_\mu\|_{\mathcal{V}} = \tilde{\beta}^{-1}_\mu \|A_\mu \hat{U}_\mu - B\|_{*\mathcal{V}'} = \tilde{\beta}^{-1}_\mu \|A_\mu U_\mu - B\|_{*\mathcal{V}'} \leq \tilde{\beta}^{-1}_\mu \|B\|_{*\mathcal{V}'}\xi = \tilde{\beta}^{-1}_\mu \delta\xi \leq \tilde{\beta}^{-1}_{\min}\delta\xi.$$

Hence, if $\tilde{\beta}^{-1}_{\min}\delta\xi \leq \text{tol}$, the validity of $\mathcal{E}_1$ and $\mathcal{E}_2$ follows from Definition 3.1. □

Since the $\|\cdot\|_{*\mathcal{V}'}$ norm is hard to compute, the stopping criterion (3.7) uses in practice the Hermitian norm in $\mathbb{C}^N$ or the $\mathcal{V}$-norm of the corresponding functions in $\mathcal{V}$.

**Remark 3.8** (Goal-oriented case, approximate reduced basis functions)**.** The formulae $\mathcal{E}_1^{\text{go}}$ and $\mathcal{E}_2^{\text{go}}$ are valid for computing the error bound with tolerance tol if $\left(\tilde{\beta}_{\min}^d\right)^{-1}\delta\gamma\xi^2 \leq \text{tol}$.

### *3.2.3. Synthesis*

Taking into account the round-off errors in the computation of the error bound and the stopping criterion of an iterative solver, and supposing that the bounds (3.3) and (3.5) are reached, the formulae $\mathcal{E}_1$ and $\mathcal{E}_2$ are valid for computing the error bound with tolerance tol if, respectively,

$$\begin{aligned} &\text{for } \mathcal{E}_1, && 2\tilde{\beta}_{\min}^{-1}\delta\max\left(\xi,\epsilon\right) \leq \text{tol},\\ &\text{for } \mathcal{E}_2, && 2\tilde{\beta}_{\min}^{-1}\delta\max\left(\xi,\sqrt{\epsilon}\right) \leq \text{tol}, \end{aligned} \tag{3.9}$$

and the formulae $\mathcal{E}_1^{\text{go}}$ and $\mathcal{E}_2^{\text{go}}$ are valid for computing the error bound with tolerance tol if, respectively,

$$\begin{aligned} &\text{for } \mathcal{E}_1^{\text{go}}, && 2\left(\tilde{\beta}_{\min}^d\right)^{-1}\delta\gamma\max\left(\xi^2,\epsilon^2\right) \leq \text{tol},\\ &\text{for } \mathcal{E}_2^{\text{go}}, && 2\left(\tilde{\beta}_{\min}^d\right)^{-1}\delta\gamma\max\left(\xi^2,\epsilon\right) \leq \text{tol}. \end{aligned} \tag{3.10}$$

Focusing on round-off errors, the formula $\mathcal{E}_1$ for computing the error bound is valid for tolerances scaling as $\epsilon$, but is not online-efficient, whereas the formula $\mathcal{E}_2$ is online-efficient but is valid only for (significantly) higher tolerances, namely tolerances scaling as $\sqrt{\epsilon}$.

## 4. New procedures for accurate and online-efficient evaluation of the error bound

In this section, online-efficient methods, that are valid for tolerances scaling as $\epsilon$, are devised to evaluate the error bound.

### 4.1. Procedure 1: rewriting $\mathcal{E}_2$

We first present the procedure proposed in [9]. We consider that a reduced basis of size $\hat{N}$ has been constructed. Let $\sigma := 1 + 2d\hat{N} + (d\hat{N})^2$. For a given $\mu \in \mathcal{P}_{\text{trial}}$ and the resulting $\hat{u}_\mu \in \text{Span}\{u_1, \ldots, u_{\hat{N}}\}$ solving the reduced problem, we define $\hat{X}(\mu) \in \mathbb{C}^\sigma$ as the vector with components $(1, \hat{x}_{\mu_I}, \hat{x}_{\mu_I}^*, \hat{x}_{\mu_I}\hat{x}_{\mu_J})$, where $\hat{x}_{\mu_I} = \alpha_k(\mu)\gamma_i(\mu)$ (we recall that $\gamma_i(\mu)$ are the coefficients of the reduced solution in the reduced basis, see (2.3), and $\alpha_k(\mu)$ the coefficients of the affine decomposition of $a_\mu$ in (2.7)), with $1 \leq I, J \leq d\hat{N}$ (with $I = i + \hat{N}(k-1)$ such that $1 \leq i \leq \hat{N}$, $1 \leq k \leq d$, and with $J = j + \hat{N}(l-1)$ such that $1 \leq j \leq \hat{N}$, $1 \leq l \leq d$). We can write the right-hand side of (2.11) as a linear form in $\hat{X}(\mu)$ as follows:

$$\delta^2 + 2\text{Re}(s^t\hat{x}_\mu) + \hat{x}_\mu^{*t}S\hat{x}_\mu = \sum_{p=1}^{\sigma} t_p\hat{X}_p(\mu), \tag{4.1}$$

where $t_p$ is independent of $\mu$ (as $\delta$, $s$, and $S$ are independent of $\mu$) and $\hat{X}_p(\mu)$ is the $p$th component of $\hat{X}(\mu)$.

Now, in the offline stage, we take $\sigma$ values (*e.g.* random values) $\mu_r \in \mathcal{P}_{\text{trial}}$, $r \in \{1, \ldots, \sigma\}$, of the parameter $\mu$. Then, we compute the vectors $\hat{X}(\mu_r)$ and the quantities

$$V_r := \sum_{p=1}^{\sigma} t_p\hat{X}_p(\mu_r). \tag{4.2}$$

Finally, we define $T \in \mathbb{C}^{\sigma\times\sigma}$ as the matrix whose columns are formed by the vectors $\hat{X}(\mu_r)$, that is, $T_{pr} = \hat{X}_p(\mu_r)$ for all $1 \leq p, r \leq \sigma$. We assume that $T$ is invertible, which always happens to be the case in our simulations.

Now, suppose that in the online stage we want to evaluate the error bound for the RB solution $\hat{u}_\mu$ computed at a certain parameter $\mu \in \mathcal{P}_{\text{trial}}$. Then, we evaluate the vector $\hat{X}(\mu)$ and solve the linear system

$$T\lambda(\mu) = \hat{X}(\mu), \tag{4.3}$$

yielding $\lambda(\mu) \in \mathbb{C}^\sigma$. We then obtain $\hat{X}(\mu) = \sum_{r=1}^{\sigma} \lambda_r(\mu)\hat{X}(\mu_r)$ and

$$\sum_{p=1}^{\sigma} t_p \hat{X}_p(\mu) = \sum_{p,r=1}^{\sigma} t_p \lambda_r(\mu)\hat{X}_p(\mu_r) = \sum_{r=1}^{\sigma} \lambda_r(\mu) V_r. \tag{4.4}$$

This yields the following new formula for computing the error bound:

$$\mathcal{E}_3(\mu) := \tilde{\beta}_\mu^{-1} \left( \sum_{r=1}^{\sigma} \lambda_r(\mu) V_r \right)^{\frac{1}{2}}, \tag{4.5}$$

where the quantities $V_r = \|G_{\mu_r}\hat{u}_{\mu_r}\|_{\mathcal{V}}^2$ can be precomputed. Thus, computing $\mathcal{E}_3$ requires solving (4.3) and summing the $\sigma$ precomputed quantities $V_r$. Since the complexity of this procedure is independent of $N$, the formula $\mathcal{E}_3$ is online-efficient for computing the error bound.

**Remark 4.1** (Goal-oriented case)**.** For the goal-oriented case, the procedure is carried out independently on the two multivariate polynomials $\|G_\mu \hat{u}_\mu\|_{\mathcal{V}}^2$ and $\|G_\mu^d \hat{v}_\mu\|_{\mathcal{V}}^2$.

Notice that $\mathcal{E}_1(\mu)$, $\mathcal{E}_2(\mu)$, and $\mathcal{E}_3(\mu)$ are equal in exact arithmetic. As pointed out in [9], the matrix $T$ exhibits in practice large condition numbers, and there is no guarantee that $T$ is actually invertible. We will see in Section 5 for a three-dimensional acoustic scattering problem that $\mathcal{E}_3$ can be in practice as ill-behaved as $\mathcal{E}_2$. Moreover, there is no *a priori* method for selecting the parameters $\mu_r$ for which the quantities $V_r$ are precomputed. In the next section, we propose a new procedure that solves these problems.

### 4.2. Procedure 2: improvement on Procedure 1 using the EIM

In the formula $\mathcal{E}_3$, a potentially ill-conditioned problem $T\lambda(\mu) = \hat{X}(\mu)$ is solved in order to exactly represent $\hat{X}(\mu)$ by the linear combination $\sum_{r=1}^{\sigma} \lambda_r(\mu)\hat{X}(\mu_r)$. Following a suggestion by Patera [33], we propose to approximate $\hat{X}(\mu)$ by means of an interpolation procedure. We want to modify the formula $\mathcal{E}_3$ by an interpolation formula relying on a better conditioned linear system. The price to pay is that the new formula $\mathcal{E}_4$ will not be equal to $\mathcal{E}_1$ in exact arithmetic; the interpolation errors are however marginal, as further discussed in Remark 4.7. We also look for a way to choose the parameters $\mu_r$ for which the quantities $V_r$ have to be precomputed. We refer to these values for $\mu_r$ as "interpolation points", and to the set of these points as $\mathcal{P}_{\text{inter}}$.

Consider the function of two variables $(p, \mu) \mapsto \hat{X}_p(\mu)$, for all $p \in \{1, \ldots, \sigma\}$ and all $\mu \in \mathcal{P}_{\text{trial}}$. We look for an approximation of this function in the form

$$\forall \mu \in \mathcal{P}_{\text{trial}}, \forall p \in \{1, \ldots, \sigma\},\ \hat{X}_p(\mu) \approx \sum_{r=1}^{\hat{\sigma}} \lambda_r^{\hat{\sigma}}(\mu)\hat{X}_p(\mu_r), \tag{4.6}$$

for a certain parameter $\hat{\sigma} \leq \sigma$. The empirical interpolation method (EIM) (more precisely the discrete EIM since $p$ is a discrete variable) provides a numerical procedure to construct this approximation and to choose the interpolation points (see [3, 30]).

For completeness, we briefly describe the EIM and adapt the notation of [30] to the present context. The EIM is an offline-online procedure. During the offline stage, $\hat{\sigma}$ basis functions are computed, denoted

$q_j : \mathcal{P}_{\text{trial}} \ni \mu \mapsto q_j(\mu) \in \mathbb{C}$, for all $j \in \{1, \dots, \hat{\sigma}\}$. These basis functions will be used in the online stage to carry out the interpolation. We define $q^{\hat{\sigma}}$ as the vector-valued map $\mathcal{P}_{\text{trial}} \ni \mu \mapsto q^{\hat{\sigma}}(\mu) := (q_j(\mu))_{1\leq j\leq \hat{\sigma}} \in \mathbb{C}^{\hat{\sigma}}$. During the offline stage, $\hat{\sigma}$ interpolation points $\mu_r \in \mathcal{P}_{\text{trial}}$ are also selected; these points are collected in the set $\mathcal{P}_{\text{inter}}$. Notice that $\mathcal{P}_{\text{select}}$, the set of parameter values selected by the greedy algorithm of the RB method, is different from $\mathcal{P}_{\text{inter}}$. During the online stage, the matrix $B^{\hat{\sigma}} \in \mathbb{C}^{\hat{\sigma},\hat{\sigma}}$, where $B^{\hat{\sigma}}_{ij} = q_i(\mu_j)$, for $1 \leq i, j \leq \hat{\sigma}$, is constructed. Letting $\mu \in \mathcal{P}_{\text{trial}}$, we solve for $\lambda^{\hat{\sigma}}(\mu) \in \mathbb{C}^{\hat{\sigma}}$ such that

$$B^{\hat{\sigma}} \lambda^{\hat{\sigma}}(\mu) = q^{\hat{\sigma}}(\mu), \tag{4.7}$$

and compute the rank-$\hat{\sigma}$ interpolation operators defined as follows.

**Definition 4.2.** Let $1 \leq k \leq \hat{\sigma}$. The rank-$k$ interpolation operator $I^k$ is defined such that

$$I^k \hat{X}(\mu) := \sum_{r=1}^{k} \lambda_r^k(\mu) \hat{X}(\mu_r), \tag{4.8}$$

where $\lambda^k(\mu) \in \mathbb{C}^k$ solves $B^k \lambda^k(\mu) = q^k(\mu)$.

Equation (4.8) defines an interpolation in the sense that $I^k \hat{X}_{p_r}(\mu) = \hat{X}_{p_r}(\mu)$ for all $1 \leq r \leq k$ and all $\mu \in \mathcal{P}_{\text{trial}}$. The formula $\hat{X}_p(\mu) \approx (I^{\hat{\sigma}} \hat{X})_p(\mu)$, for all $\mu \in \mathcal{P}_{\text{trial}}$ and all $p \in \{1, \dots, \sigma\}$, provides the approximate interpolation formula searched for in (4.6).

**Definition 4.3.** The residual operator $\delta^{\hat{\sigma}}$ is defined by

$$\delta^{\hat{\sigma}} := \text{Id} - I^{\hat{\sigma}}. \tag{4.9}$$

Algorithm 1 presents the construction of the function $q^{\hat{\sigma}}$ by a greedy algorithm during the offline stage.

**Algorithm 1** Offline stage of the EIM

```
Choose $\hat{\sigma} > 1$    [Number of interpolation points]
Set $k := 1$
Compute $p_1 := \underset{p\in\{1,\dots,\sigma\}}{\text{argmax}} \|\hat{X}_p(\cdot)\|_{\ell^\infty(\mathcal{P}_{\text{trial}})}$
Compute $\mu_1 := \underset{\mu\in\mathcal{P}_{\text{trial}}}{\text{argmax}} |\hat{X}_{p_1}(\mu)|$ and set $\mathcal{P}_{\text{inter}} = \{\mu_1\}$    [First interpolation point]
Set $q_1(\cdot) := \dfrac{\hat{X}_{p_1}(\cdot)}{\hat{X}_{p_1}(\mu_1)}$    [First basis function]
Set $B^1_{11} := 1$    [Initialize $B$ matrix]
while $k < \hat{\sigma}$ do
    Compute $p_{k+1} := \underset{p\in\{1,\dots,\sigma\}}{\text{argmax}} \|(\delta^k \hat{X})_p(\cdot)\|_{\ell^\infty(\mathcal{P}_{\text{trial}})}$
    Compute $\mu_{k+1} := \underset{\mu\in\mathcal{P}_{\text{trial}}}{\text{argmax}} |(\delta^k \hat{X})_{p_{k+1}}(\mu)|$    [$(k+1)$-th interpolation point]
    Set $\mathcal{P}_{\text{inter}} := \mathcal{P}_{\text{inter}} \cup \{\mu_{k+1}\}$    [Update of $\mathcal{P}_{\text{inter}}$]
    Set $q_{k+1}(\cdot) := \dfrac{(\delta^k \hat{X})_{p_{k+1}}(\cdot)}{(\delta^k \hat{X})_{p_{k+1}}(\mu_{k+1})}$    [$(k+1)$-th basis function]
    $B^{k+1}_{ij} := q_j(\mu_i)$, $1 \leq i, j \leq k+1$    [$(k+1)$-th $B$ matrix]
    $k \leftarrow k + 1$    [Increment the size of the interpolation]
end while
```

**Definition 4.4.** The new formula for computing the error bound is

$$\mathcal{E}_4(\mu) := \tilde{\beta}_\mu^{-1}\left(\sum_{r=1}^{\hat{\sigma}} \lambda_r^{\hat{\sigma}}(\mu) V_r\right)^{\frac{1}{2}}, \tag{4.10}$$

where $\lambda^{\hat{\sigma}}(\mu)$ is the solution to (4.7). We recall that $V_r = \|G_{\mu_r}\hat{u}_{\mu_r}\|_{\mathcal{V}}^2$.

**Proposition 4.5.** *The computation of the formula $\mathcal{E}_4$ is well defined, and this formula is online-efficient.*

*Proof.* Owing to [30], Theorem 1, the matrix $B$ is lower triangular with diagonal unity. Hence, $\det B = 1$ and $B$ is guaranteed to be invertible. The online procedure of EIM, consisting in solving a linear system defined by the matrix $B$, is thus well defined. Then, since the EIM procedure in carried out on $\hat{X}_p(\mu)$, for all $p \in \{1, \ldots, \sigma\}$ and all $\mu \in \mathcal{P}_{\text{trial}}$, all the computations involved are of complexity independent of $N$, even the offline part of the EIM. Finally, the complexity of the online part of EIM only depends on $\hat{\sigma}$. □

**Remark 4.6** (Stopping criterion in Algorithm 1)**.** For ease of presentation, we chose a simple stopping criterion based on an *a priori* fixed maximum number of interpolation points. In practice, one possibility is to stop the algorithm when the maximal approximation error in the EIM is below a prescribed value, by monitoring the quantity $(\delta^k \hat{X})_{p_{k+1}}(\mu_{k+1})$.

**Remark 4.7** (Interpolation errors)**.** As already observed, $\mathcal{E}_4$ does not equal $\mathcal{E}_1$ in exact arithmetics owing to interpolation errors (when $\hat{\sigma} < \sigma$). Thus, although Algorithm 1 yields an accurate approximation of $\hat{X}_p(\mu)$, a given interpolation error on $\hat{X}_p(\mu)$ does not directly translate into a bound on the difference between $\mathcal{E}_1(\mu)$ and $\mathcal{E}_4(\mu)$ (the latter depending also on $\delta$, $s$, and $S$, as well as on $\tilde{\beta}_\mu$). We observe in our numerical experiments that these latter errors are lower than the errors incurred in the evaluation of $\mathcal{E}_2$ (due to round-off errors) and in the evaluation of $\mathcal{E}_3$ (due to the poor conditioning of $T$).

## 4.3. Illustration

Consider as in [9] a one-dimensional linear diffusion problem, namely the boundary value problem $-u'' + \mu u = 1$ on $]0,1[$ with $u(0) = u(1) = 0$, with parameter $\mu \in \mathcal{P} := [1, 100]$. The analytic solution is

$$u(x) = -\frac{1}{\mu}\left(\cosh\left(\sqrt{\mu}x\right) - 1\right) + \frac{\cosh\left(\sqrt{\mu}\right) - 1}{\mu \sinh\left(\sqrt{\mu}\right)} \sinh\left(\sqrt{\mu}x\right). \tag{4.11}$$

The Lax–Milgram theory is valid, and the coercivity constant is bounded from below by 1 in the $H^1$-norm. The error bound is given by $\mathcal{E}_1(\mu) = \|G_\mu \hat{u}_\mu\|_{H^1(]0,1[)}$. Lagrange $\mathbb{P}_1$ finite elements are used with uniform mesh cells of length 0.005. The set $\mathcal{P}_{\text{trial}}$ consists of 1000 points uniformly distributed in $\mathcal{P}$. The RB method is carried out until the formula $\mathcal{E}_2$ suffers from round-off errors, which already happens for a reduced basis of size $\hat{N} = 7$ (since $d = 2$, we obtain $\sigma = 225$). A direct solver is used, so that the only adverse phenomenon to compute the error bound are round-off errors.

In Figure 2, we see that the classical formula $\mathcal{E}_2$ is not valid for computing the error bound with any tolerance below $10^{-7}$, whereas the formulae $\mathcal{E}_1$, $\mathcal{E}_3$ and $\mathcal{E}_4$ are valid with tolerances down to $10^{-14}$. The difference is of 7 orders of magnitude; given that $\sqrt{\epsilon} \approx 10^{-7}$, this is consistent with Remark 3.3 and Section 4.1.

In Figure 3, we observe that instabilities occur in the formula $\mathcal{E}_3$, especially for parameter values close to the elements of $\mathcal{P}_{\text{select}}$. This is due to the poor conditioning of the matrix $T$ when solving (4.3). The new formula $\mathcal{E}_4$ based on the EIM is seen to introduce much less numerical errors than $\mathcal{E}_3$.

In Figure 4, we plot $\max\limits_{\mu \in \mathcal{P}_{\text{select}}} (\mathcal{E}_4(\mu))$ as a function of $\hat{\sigma}$. From this figure and Definition 3.1, we deduce that for $\hat{\sigma} \geq 23$, the formula $\mathcal{E}_4$ is valid for any tolerance larger than $10^{-12}$. If we want to consider a tolerance of the order of $10^{-14}$, we need $\hat{\sigma} > 23$.

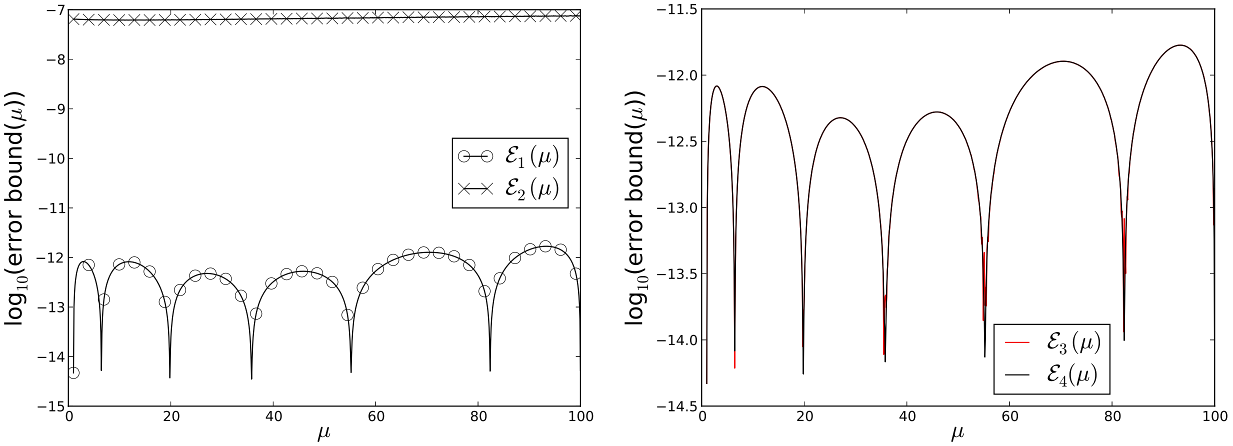

FIGURE 2. Error bound curves with respect to the parameter. The formula $\mathcal{E}_4$ is computed with $\hat{\sigma} = 23$.

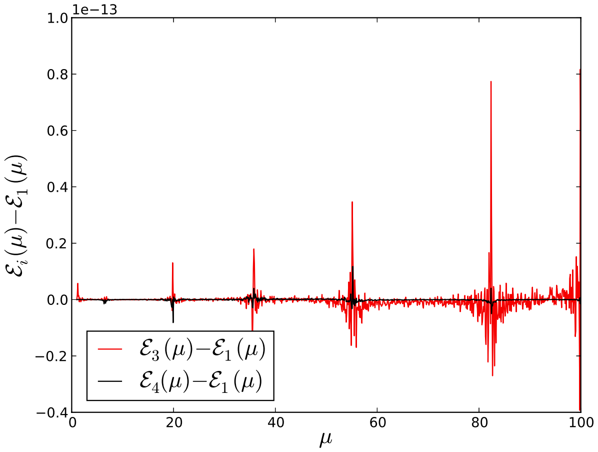

FIGURE 3. Comparison of the formulae $\mathcal{E}_3$ and $\mathcal{E}_4$, with respect to the formula $\mathcal{E}_1$.

### 4.4. Procedure 3: improvement of Procedure 2 using a stabilized EIM

In practice, round-off errors are accumulated during the loop in Algorithm 1, and if we keep increasing the number of interpolation points, the coefficients of the matrix $B$ suffer from round-off errors, so that the relation $\det(B) = 1$ no longer holds. The matrix $B$ becomes non invertible at some stage. To solve this problem, we now propose a numerical stabilization of EIM based on the following property:

**Property 4.8.** *There holds*

$$\forall i < j,\ I^j \circ I^i = I^i, \tag{4.12}$$

*where the interpolation operators $I^j$ are defined by* (4.8).

*Proof.* Using [30], Lemma 1, $I^i\hat{X} \in \mathrm{Span}\,(q_1, \ldots, q_i)$ and $I^i v = v$ for all $v \in \mathrm{Span}\,(q_1, \ldots, q_i)$. Therefore, $I^j \circ I^i\hat{X} = I^i\hat{X}$ for all $i < j$. □

In our numerical experiments, we observe that, as the number of iterations of the greedy procedure for the EIM grows, the relation (4.12) is no longer verified numerically, due to accumulation of round-off errors. These

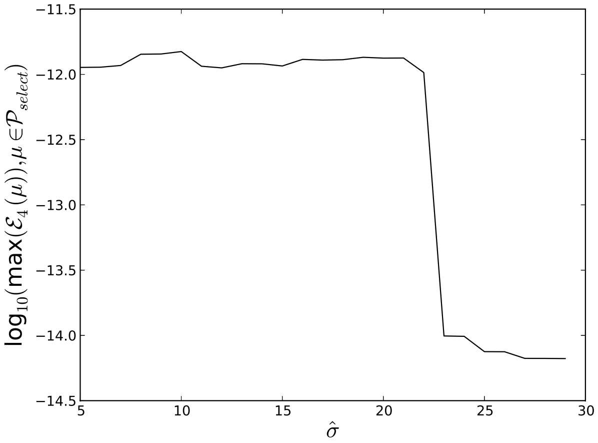

FIGURE 4. $\max\limits_{\mu\in\mathcal{P}_{\text{select}}}(\mathcal{E}_4(\mu))$ as a function of $\hat\sigma$.

TABLE 1. Comparison between stabilized Gram–Schmidt and stabilized EIM.

| | stabilized Gram–Schmidt | stabilized EIM |
|---|---|---|
| global input | $(v_1,\dots,v_{\hat\sigma})$ basis of $\mathbb{C}^{\hat\sigma}$ | $v:\mathcal{P}_{\text{trial}}\to\mathbb{C}^{\hat\sigma}$ |
| classical residual at step $k$ | $\delta^k v_k = v_k - \Pi^k v_k$ | $(\delta^k v)(\mu) = v(\mu) - (I^k v)(\mu)$ |
| intermediate residuals at step $k$ | $\delta^{k,1}_{\text{stab}}v_k = v_k - \Pi^1 v_k$<br>$\delta^{k,2}_{\text{stab}}v_k = \delta^{k,1}_{\text{stab}}v_k - \Pi^2\delta^{k,1}_{\text{stab}}v_k,$<br>$\vdots$<br>$\delta^{k,k}_{\text{stab}}v_k = \delta^{k,k-1}_{\text{stab}}v_k - \Pi^k\delta^{k,k-1}_{\text{stab}}v_k$ | $(\delta^{k,1}_{\text{stab}}v)(\mu) = v(\mu) - (I^1 v)(\mu)$<br>$(\delta^{k,2}_{\text{stab}}v)(\mu) = (\delta^{k,1}_{\text{stab}}v)(\mu) - I^2(\delta^{k,1}_{\text{stab}}v)(\mu),$<br>$\vdots$<br>$(\delta^{k,k}_{\text{stab}}v)(\mu) = (\delta^{k,k-1}_{\text{stab}}v)(\mu) - I^k(\delta^{k,k-1}_{\text{stab}}v)(\mu)$ |
| stabilized residual at step $k$ | $\delta^k_{\text{stab}}v_k = \delta^{k,k}_{\text{stab}}v_k$ | $(\delta^k_{\text{stab}}v)(\mu) = (\delta^{k,k}_{\text{stab}}v)(\mu)$ |
| global output | $(\delta^1_{\text{stab}}v_1,\delta^2_{\text{stab}}v_2,\dots,\delta^{\hat\sigma}_{\text{stab}}v_{\hat\sigma})$<br>orthogonal basis of $\text{Span}(v_1,\dots,v_{\hat\sigma})$ | $(I^{\hat\sigma}v)(\mu)$ |

numerical instabilities can be compensated in the same fashion as the Gram–Schmidt orthonormalization procedure is stabilized (see [22], Chap. 5.2.8). The Gram–Schmidt algorithm transforms a linearly independent family of vectors $\{v_i\}$ into an orthonormal basis $\{u_i\}$. To simplify the presentation, we suppose in what follows that the normalization step is not carried out. Consider the orthogonalization step for the $k$th vector. We denote by $\Pi^k$ the projection operator on $\text{Span}(u_1,\dots,u_k)$, and $\delta^k := \text{Id} - \Pi^k$. For the EIM, we suppose that $(k-1)$ interpolation operators $I^i$, $1\leq i\leq k-1$, have been constructed, and we wish to construct the $k$th interpolation operator $I^k$. A comparison between the stabilized Gram–Schmidt orthonormalization procedure and the proposed stabilization for the EIM is presented in Table 1.

**Proposition 4.9.** *Let $k\in\mathbb{N}^*$. In exact arithmetic, the following relations hold for the residuals defined in Table 1: $\delta^k_{\text{stab}}v = \delta^k v$.*

*Proof.* We prove by recursion that, for all $i\leq k$, $\delta^{k,i}_{\text{stab}} = \delta^i$. The case $i=1$ is clear from the definition of the first intermediate residual in Table 1. Let $i\leq k$ and suppose that $\delta^{k,i-1}_{\text{stab}} = \text{Id} - I^{i-1}$ for the EIM. There holds

$$\delta^{k,i}_{\text{stab}} = \delta^{k,i-1}_{\text{stab}} - I^i\circ\delta^{k,i-1}_{\text{stab}} = \text{Id} - I^{i-1} - I^i + I^i\circ I^{i-1} = \text{Id} - I^i = \delta^i, \tag{4.13}$$

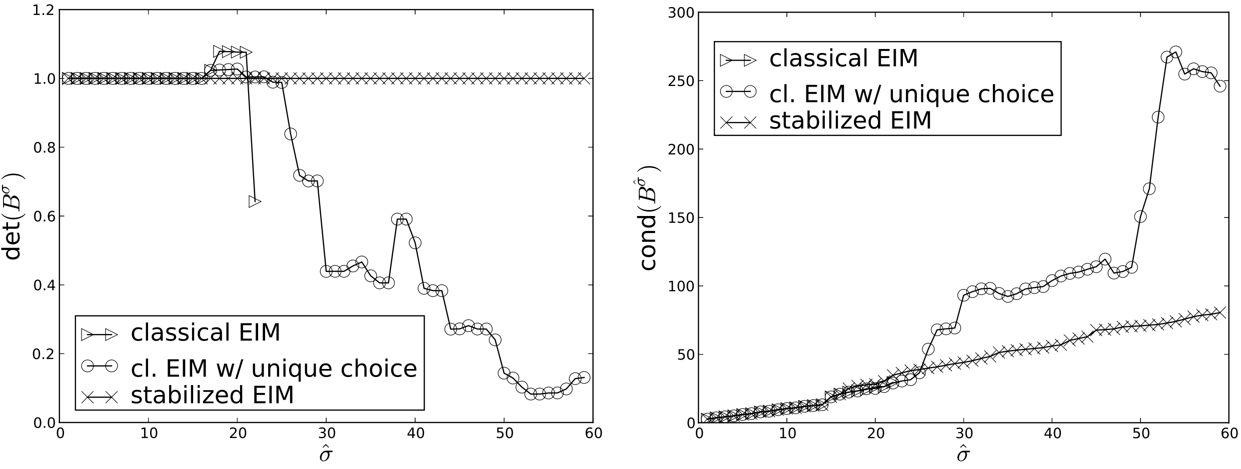

FIGURE 5. Determinant (left) and condition number (right) of the matrix $B^{\hat{\sigma}}$ as a function of $\hat{\sigma}$, for the classical EIM, the classical EIM with unique choice, and the stabilized EIM. The classical EIM curves stop at 21 interpolation points since $B^{\hat{\sigma}}$ becomes non invertible at 22 points.

since $I^i \circ I^{i-1} = I^{i-1}$ owing to Property 4.8. The results follow from the case $i = k$. The same relation is proved likewise for the Gram–Schmidt procedure, for which $\Pi^i \circ \Pi^{i-1} = \Pi^{i-1}$ holds as well. □

**Definition 4.10** (Stabilized EIM)**.** The stabilized EIM consists in the same offline procedure as the one described in Section 4.2, except that the residuals $\delta^k$ are replaced by the stabilized residuals $\delta^k_{\text{stab}}$ defined in Table 1. The online stage is the same as that of the classical EIM.

The stabilized Gram–Schmidt procedure generates a set of vectors much less polluted by round-off errors (see [4,20]). By analogy we expect that the stabilized EIM produces a more accurate interpolation procedure than the classical EIM, that is, much less polluted by round-off errors. This is numerically verified in Figure 5, where $\det(B^{\hat{\sigma}})$ and $\text{cond}(B^{\hat{\sigma}})$ are represented as a function of $\hat{\sigma}$. We consider the test case described in Section 4.3, where we recall that $\hat{N} = 7$, $d = 2$, and $\sigma = 225$. If the method is stable, then $\det(B^{\hat{\sigma}}) = 1$ should hold throughout the process. Figure 5 shows that the stabilized EIM behaves as intended. The classical EIM curve stops since the matrix $B^{\hat{\sigma}}$ becomes noninvertible at some point: a parameter already in $\mathcal{P}_{\text{inter}}$ has been selected by the greedy algorithm. Invertibility can be recovered artificially by ensuring that the new interpolation point is not an element of the current set $\mathcal{P}_{\text{inter}}$. We call this procedure EIM with unique choice. However, this fix is not completely satisfactory, since $\det(B^{\hat{\sigma}}) = 1$ is not satisfied. Moreover, $\text{cond}(B^{\hat{\sigma}})$ is much more ill-behaved with this procedure than with the stabilized EIM.

**Remark 4.11** (Computational cost and variant of stabilized EIM)**.** The computational cost of the stabilized EIM is more than that of the classical EIM, since the stabilized residual requires as many calls to a classical residual as the number of selected interpolation points (*i.e.* the scaling with $\hat{\sigma}$ is $\hat{\sigma}^2$ for the stabilized EIM as opposed to $\hat{\sigma}$ for the classical EIM). One can think of a cheaper procedure by monitoring $\det(B^{\hat{\sigma}})$ and adding some intermediate residuals $\delta^{k,j}_{\text{stab}}$ until $\det(B^{\hat{\sigma}})$ is close enough to 1.

## 4.5. Summary

The advantages and drawbacks of the four considered formulae for computing the error bound are summarized in Table 2. To estimate the computational complexity of the methods, we keep only the leading order in operation count. We denote the complexity of the resolution of (2.12) by $N_{\text{sol}}$. The linear systems of size $\sigma$, $\hat{\sigma}$, and $\hat{N}$ are supposed to be solved by a direct solver, hence with complexity proportional to $\sigma^3$, $\hat{\sigma}^3$, and $\hat{N}^3$, respectively.

TABLE 2. Comparison of the considered formulae for computing the error bound.

| Property | $\mathcal{E}_1$ | $\mathcal{E}_2$ | $\mathcal{E}_3$ | $\mathcal{E}_4$ |
|---|---|---|---|---|
| Online efficient | No | Yes | Yes | Yes |
| Unconditionally well-posed | Yes | Yes | No | Yes |
| Dependence on $\epsilon$ of the observed accuracy | $\epsilon$ | $\sqrt{\epsilon}$ | $\epsilon$, if well-posed | $\epsilon$ |
| Equals $\mathcal{E}_1$ in exact arithmetics | – | Yes | Yes | Yes, if $\hat{\sigma} = \sigma$<br>No, if $\hat{\sigma} < \sigma$ |
| Complexity of the offline stage | – | $(d\hat{N}+1)N_{\rm sol}$ | $\sigma N_{\rm sol}$ | $\hat{\sigma}^4\sigma M + \hat{\sigma}N_{\rm sol}$ with classical EIM<br>$\hat{\sigma}^5\sigma M + \hat{\sigma}N_{\rm sol}$ with stabilized EIM |
| Complexity of the online stage | – | $\hat{N}^3 + \sigma$ | $\hat{N}^3 + \sigma^3$ | $\hat{N}^3 + \hat{\sigma}^3$ |

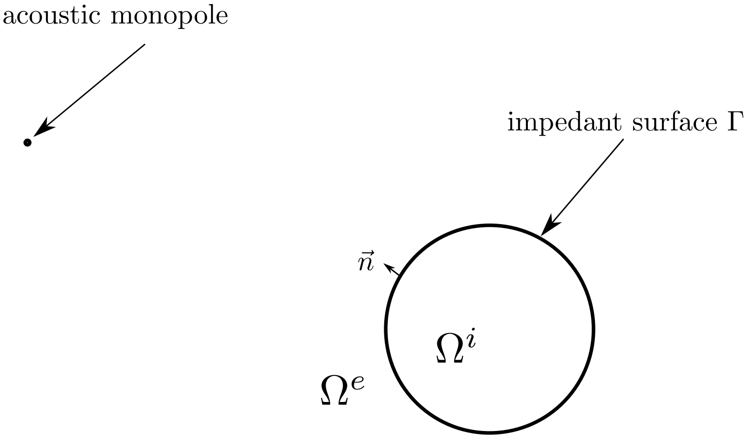

FIGURE 6. Geometry for the three-dimensional acoustic scattering problem.

For the offline stage of $\mathcal{E}_2$ and $\mathcal{E}_3$, we have to evaluate respectively $d\hat{N}+1$ and $\sigma$ times the functional $G_\mu$, which requires to solve (2.12). For the offline stage of $\mathcal{E}_4$, let $M$ denote the cardinality of $\mathcal{P}_{\rm trial}$. The $k$-loop in Algorithm 1 requires at each step to compute a maximum over $\sigma$ different $\ell^\infty(\mathcal{P}_{\rm trial})$ norms, and then to solve a linear system of size $k$, leading to a complexity of $\hat{\sigma}^4\sigma M + \hat{\sigma}N_{\rm sol}$. If the stabilized EIM is used instead for $\mathcal{E}_4$, each residual evaluation in the $k$-loop requires solving $k$ linear systems of size 1 to $k$, leading to a complexity of $\hat{\sigma}^5\sigma M + \hat{\sigma}N_{\rm sol}$. For the online stage, all the formulae require to solve the problem $\hat{E}_\mu$ of size $\hat{N}$. Moreover, $\mathcal{E}_2$ additionally requires a linear combination of size $\sigma$, whereas $\mathcal{E}_3$ and $\mathcal{E}_4$ require to solve a linear system of size $\sigma$ and $\hat{\sigma}$ respectively. We notice that if $N_{\rm sol} \gg \hat{\sigma}^4\sigma M$ and $\hat{\sigma} < d\hat{N}+1$, then the offline stage of $\mathcal{E}_4$ with stabilized EIM requires less precomputations than the offline stage of $\mathcal{E}_2$.

## 5. Application to a three-dimensional acoustic scattering problem

### 5.1. Formulation of the problem

We consider a ball $\Omega^i \subset \mathbb{R}^3$ with boundary $\Gamma$ and $\Omega^e := \mathbb{R}^3 \backslash \overline{\Omega^i}$, see Figure 6. We consider a monopole source located in $\Omega^e$. The surface of the ball is impedant, meaning that any incident wave will be partially absorbed and partially scattered. The proportion of absorbed and scattered parts is quantified by the impedance coefficient $\mu$, which is used in a Robin boundary condition at $\Gamma$. We are interested in the computation of the scattered field $p_{\rm sc}$ in $\Omega^e$. We denote $p_{\rm inc}$ the known pressure field created by the source in the absence of the sphere; the total acoustic field in $\Omega^e$ is the sum of $p_{\rm inc}$ and $p_{\rm sc}$.

We define the distribution $v : \Omega^e \cup \Omega^i \longrightarrow \mathbb{C}$ such that $v_{|\Omega^i} = -p_{\rm inc}$, $v_{|\Omega^e} = p_{\rm sc}$. We denote $\lambda$ and $\chi$ the jumps of the Neumann and Dirichlet traces of $v$ across $\Gamma$. The Robin boundary condition writes $\lambda + \frac{ik}{\mu}\chi = 0$. Since $v$ solves the homogeneous Helmholtz equation in $\Omega^e$ and in $\Omega^i$ and satisfies the Sommerfeld radiation condition at infinity, there holds

$$v = -\mathcal{S}\lambda + \mathcal{D}\chi \quad \text{in } \Omega^e \cup \Omega^i, \tag{5.1}$$

where $\mathcal{S}$ and $\mathcal{D}$ are respectively the single- and double-layer potentials. Taking the interior Dirichlet and Neumann traces of $v$ in equation (5.1) and injecting the Robin boundary condition, we obtain

$$\begin{bmatrix} N - \frac{ik}{2\mu} I & \tilde{D} \\ D & -S - \frac{i\mu}{2k} I \end{bmatrix} \begin{bmatrix} \chi \\ \lambda \end{bmatrix} = \begin{bmatrix} \gamma_1^- p_{\rm inc} \\ -\gamma_0^- p_{\rm inc} \end{bmatrix}, \tag{5.2}$$

where $k$ is the wave number of the monopole source, $N$, $\tilde{D}$, $D$ and $S$ are classical boundary integral operators (see [37]), and $\gamma_0^- p_{\rm inc}$ and $\gamma_1^- p_{\rm inc}$ are respectively the interior Dirichlet and Neumann traces of the known function $p_{\rm inc}$. Solving one of these two equations, together with the Robin boundary condition, is sufficient. The software we are using, ACTIPOLE (see [16, 17]), deals with the block system defined in (5.2), which presents the advantage of being invertible for all frequencies of the source, when the surface $\Gamma$ is Lipschitz. We denote $A_\mu$ the block operator defined by the left-hand side of (5.2). From [26, 31, 37], we infer that $A_\mu$ is a bounded bijective operator from $H^{\frac{1}{2}}(\Gamma) \times L^2(\Gamma)$ into $H^{-\frac{1}{2}}(\Gamma) \times L^2(\Gamma)$ (see also [10]). The variational form is as follows: find $(\chi, \lambda) \in H^{\frac{1}{2}}(\Gamma) \times L^2(\Gamma)$ such that for all $(\hat{\chi}, \hat{\lambda}) \in H^{\frac{1}{2}}(\Gamma) \times L^2(\Gamma)$,

$$\begin{cases} \left(N\chi - \dfrac{ik}{2\mu}\chi, \hat{\chi}\right) + \left(\tilde{D}\lambda, \hat{\chi}\right) = \left(\gamma_1 p_{\rm inc}, \hat{\chi}\right), \\ \left\langle \hat{\lambda}, D\chi \right\rangle - \left\langle \hat{\lambda}, S\lambda + \dfrac{i\mu}{2k}\lambda \right\rangle = -\left\langle \hat{\lambda}, \gamma_0 p_{\rm inc} \right\rangle, \end{cases} \tag{5.3}$$

where $(\cdot,\cdot)$ denotes the $H^{\frac{1}{2}}(\Gamma) \times H^{-\frac{1}{2}}(\Gamma)$ duality product and $\langle \cdot,\cdot \rangle$ denotes the $L^2(\Gamma)$ inner product.

Let $\mathcal{M}$ be a shape-regular triangular mesh of $\Gamma$ with meshsize $h$, and let $V_h^1$ and $V_h^0$ be respectively the spaces spanned by continuous piecewise affine polynomials on $\mathcal{M}$ and piecewise constant polynomials on $\mathcal{M}$. Let $(\phi_i)_{1 \le i \le P}$ and $(\psi_j)_{1 \le j \le P'}$ be the usual bases of $V_h^1$ and $V_h^0$ of size $P$ and $P'$, respectively. The product space $V_h^1 \times V_h^0$ is a conforming approximation of $H^{\frac{1}{2}}(\Gamma) \times L^2(\Gamma)$. The discrete problem is derived from a Galerkin procedure on $V_h^1 \times V_h^0$ using the boundary element method (BEM). From [26], the obtained discrete approximation of the problem (5.3) is inf-sup stable for $h$ small enough (see also [10]). A direct solver is used, in double-precision format.

## 5.2. Application of the RB method

The RB method has recently been applied to problems solved by means of integral equations in electromagnetism, see [13, 19]. In these works, the classical *a posteriori* error bounds were used. We are here interested in the application of our improved *a posteriori* error bounds to such problems. We take as parameter for the RB method the value of the impedance $\mu$, which is supposed here to be a positive real number. To recover an affine dependence on the parameter $\mu$, we write the BEM matrix in the form $A_\mu = a_1(\mu)A_1 + a_2(\mu)A_2 + a_3(\mu)A_3$, so that $d = 3$ in the affine decomposition (2.7) with $a_1(\mu) = 1$, $a_2(\mu) = \frac{1}{\mu}$ and $a_3(\mu) = \mu$. Specifically,

$$A_1 = \left[\begin{array}{c|c} \left(N\phi_i, \phi_j\right)_{\substack{1 \le i \le P \\ 1 \le j \le P}} & \left(\tilde{D}\psi_j, \phi_i\right)_{\substack{1 \le i \le P \\ 1 \le j \le P'}} \\ \hline \left\langle D\phi_j, \psi_i \right\rangle_{\substack{1 \le i \le P' \\ 1 \le j \le P}} & \left\langle -S\psi_i, \psi_j \right\rangle_{\substack{1 \le i \le P' \\ 1 \le j \le P'}} \end{array}\right], \tag{5.4}$$

$$A_2 = \left[\begin{array}{c|c} -\frac{ik}{2}\left(\phi_i, \phi_j\right)_{\substack{1 \le i \le P \\ 1 \le j \le P}} & (0)_{\substack{1 \le i \le P \\ 1 \le j \le P'}} \\ \hline (0)_{\substack{1 \le i \le P' \\ 1 \le j \le P}} & (0)_{\substack{1 \le i \le P' \\ 1 \le j \le P'}} \end{array}\right], \quad A_3 = \left[\begin{array}{c|c} (0)_{\substack{1 \le i \le P \\ 1 \le j \le P}} & (0)_{\substack{1 \le i \le P \\ 1 \le j \le P'}} \\ \hline (0)_{\substack{1 \le i \le P' \\ 1 \le j \le P'}} & -\frac{i}{2k}\left\langle \psi_i, \psi_j \right\rangle_{\substack{1 \le i \le P' \\ 1 \le j \le P'}} \end{array}\right]. \tag{5.5}$$

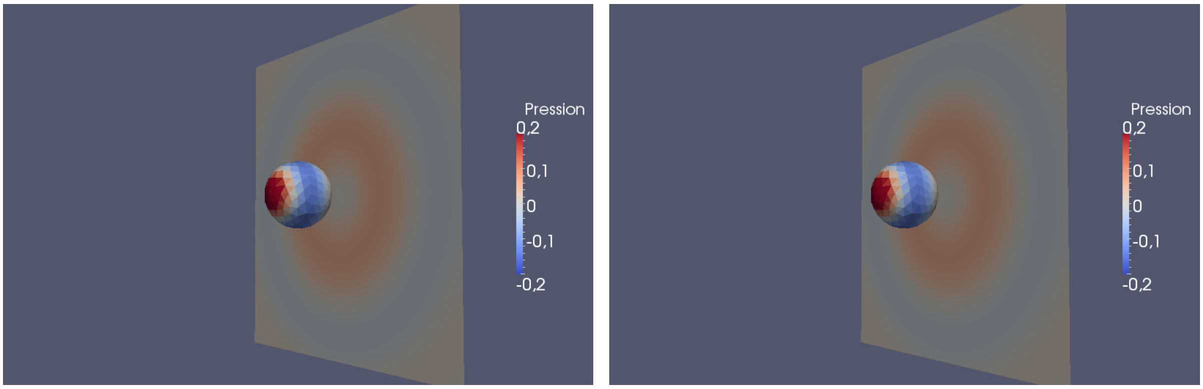

FIGURE 7. Real part of the pressure field for the BEM solution (left) and the RB solution (right), with a basis of size 10. The difference between the two fields is less than $10^{-15}$ in infinity norm.

In the general-purpose RB, the quantity of interest is the pair of potentials $(\chi, \lambda)$ on $\Gamma$. For the goal-oriented case, we consider the value of the pressure at a given point in $\Omega^e$. If this point is far enough from $\Gamma$, approximations can be made in the representation formula for the pressure. This is the far-field approximation, which consists in a linear form $Q$ acting on the solution pair $(\chi, \lambda)$ as

$$Q(\chi,\lambda) = \begin{pmatrix} -ik\dfrac{\mathrm{e}^{-ik\|x\|_2}}{4\pi\|x\|_2}\left(\mathrm{e}^{-iky\cdot\frac{x}{\|x\|_2}}\dfrac{x}{\|x\|_2}\cdot n(y),\chi(y)\right) \\ ik\dfrac{\mathrm{e}^{-ik\|x\|_2}}{4\pi\|x\|_2}\displaystyle\int_\Gamma\left(\mathrm{e}^{-iky\cdot\frac{x}{\|x\|_2}},\lambda(y)\right) \end{pmatrix} \in \mathbb{C}^2. \tag{5.6}$$

For simplicity, we take the Euclidian norm of vectors in $\mathbb{C}^{P+P'}$ instead of the $H^{\frac{1}{2}}(\Gamma) \times L^2(\Gamma)$ norms of the reconstructed functions. This way, the Riesz isomorphism $J$ is simply the identity. Therefore, the computation of the terms $G_\mu u_\mu$, as well as that of the terms $G_k u_i$, does not require to invert the stiffness matrix as in (2.12). The Successive Constraint Method is used to compute a lower bound of the inf-sup constant, which is around $10^{-6}$ in the present examples.

We define two test cases: (i) one impedant sphere ($d = 3$), with $N = 584$ and $\mu \in \mathcal{P} := [0.9, 1.1]$, (ii) two impedant spheres ($d = 5$), with $N = 1561$ and $\mu \in \mathcal{P} := [0.99, 1.01]^2$. We present visualizations of the scattered pressure field, at a random value of the parameter $\mu$, for test case (i) with $\#\mathcal{P}_{\text{trial}} = 100$ and $\hat{N} = 10$ in Figure 7 and for test case (ii) with $\#\mathcal{P}_{\text{trial}} = 225$ and $\hat{N} = 10$ in Figure 8.

### 5.3. Error bound curves

We present the error bound curves for test case (i) with a general-purpose RB, $\#\mathcal{P}_{\text{trial}} = 100$, $(\hat{N}, \hat{\sigma}, \sigma) = (2, 7, 49), (3, 10, 100), (4, 20, 169)$, and $(5, 30, 256)$ in Figure 9 and for test case (ii) with a goal-oriented RB, $\#\mathcal{P}_{\text{trial}} = 225$, $\hat{N} = 8$, $\hat{\sigma} = 60$, and $\sigma = 1681$ in Figure 10.

In test case (i), the classical formula $\mathcal{E}_2$ exhibits quite poor performances, since it cannot compute values below $10^{-4}$. This is explained by the values of the inf-sup constant which are around $10^{-6}$. Furthermore, in agreement with Remark 3.3, the lowest computable values of $\mathcal{E}_1$ and $\mathcal{E}_2$ differ by 8 orders of magnitude. In test case (ii), the behavior of formula $\mathcal{E}_3$ is quite poor, and we do not observe the level of accuracy we observed so far for $\mathcal{E}_3$. Here, the matrix $T$ defined in (4.3) is so ill-conditioned that the numerical errors introduced by its resolution are larger than the ones introduced by the formula $\mathcal{E}_2$. Furthermore, the formula $\mathcal{E}_4$ exhibits,

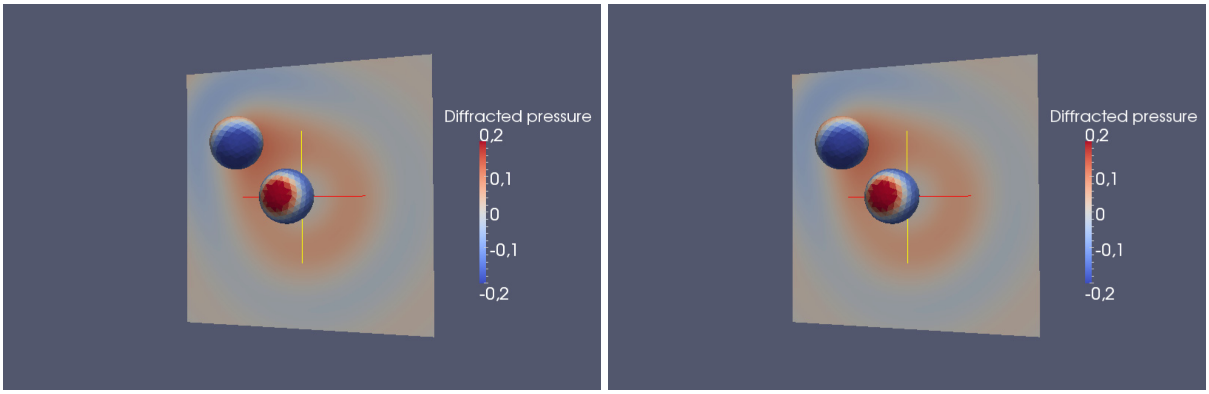

FIGURE 8. Real part of the pressure field for the BEM solution (left) and the RB solution (right), with a basis of size 10. The difference between the two fields is less than $10^{-15}$ in infinity norm.

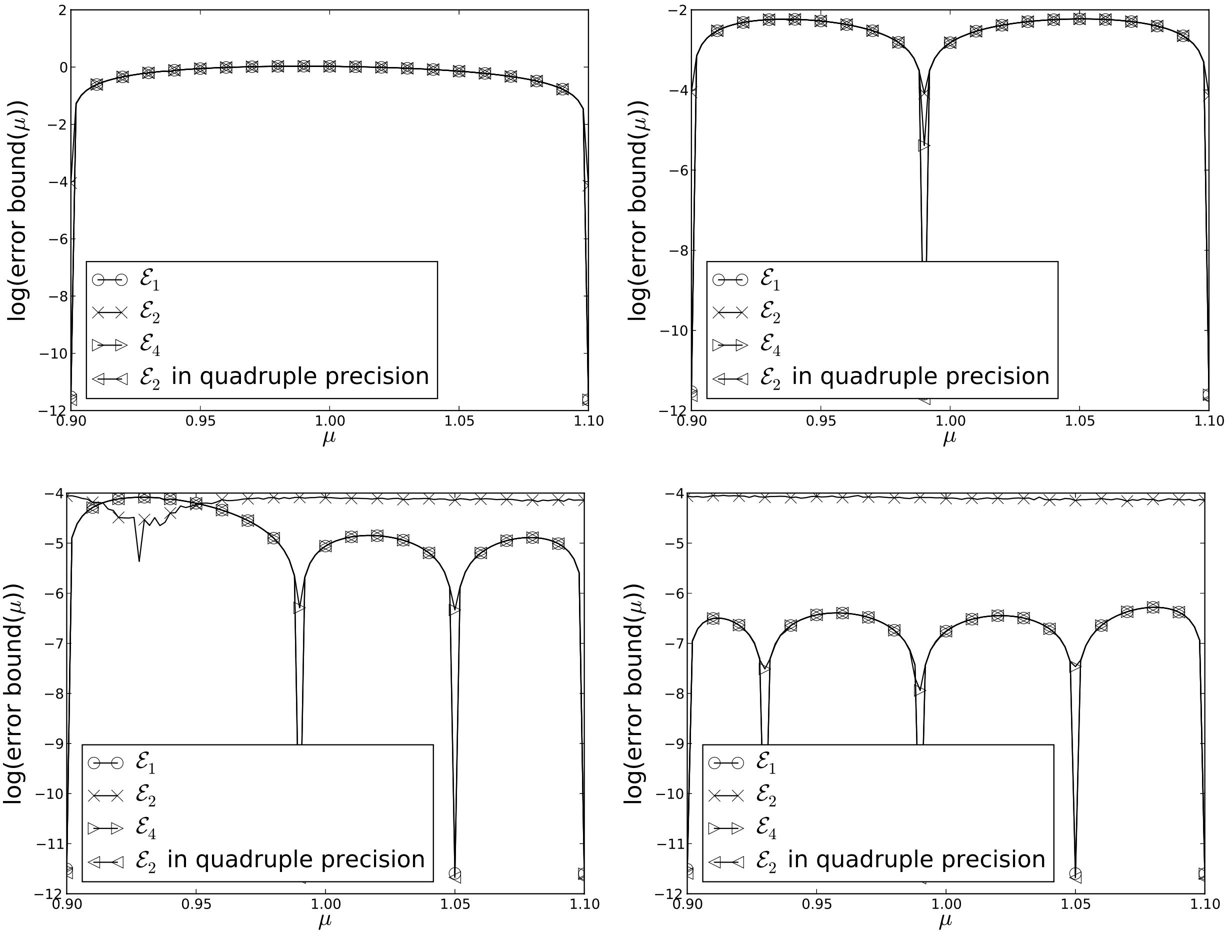

FIGURE 9. Error bound curves with respect to the impedance coefficient, with $\hat{N}$ equal to 2, 3, 4, and 5 (from left to right and top to bottom). The curve for $\mathcal{E}_2$ computed in quadruple precision superimposes to $\mathcal{E}_1$.

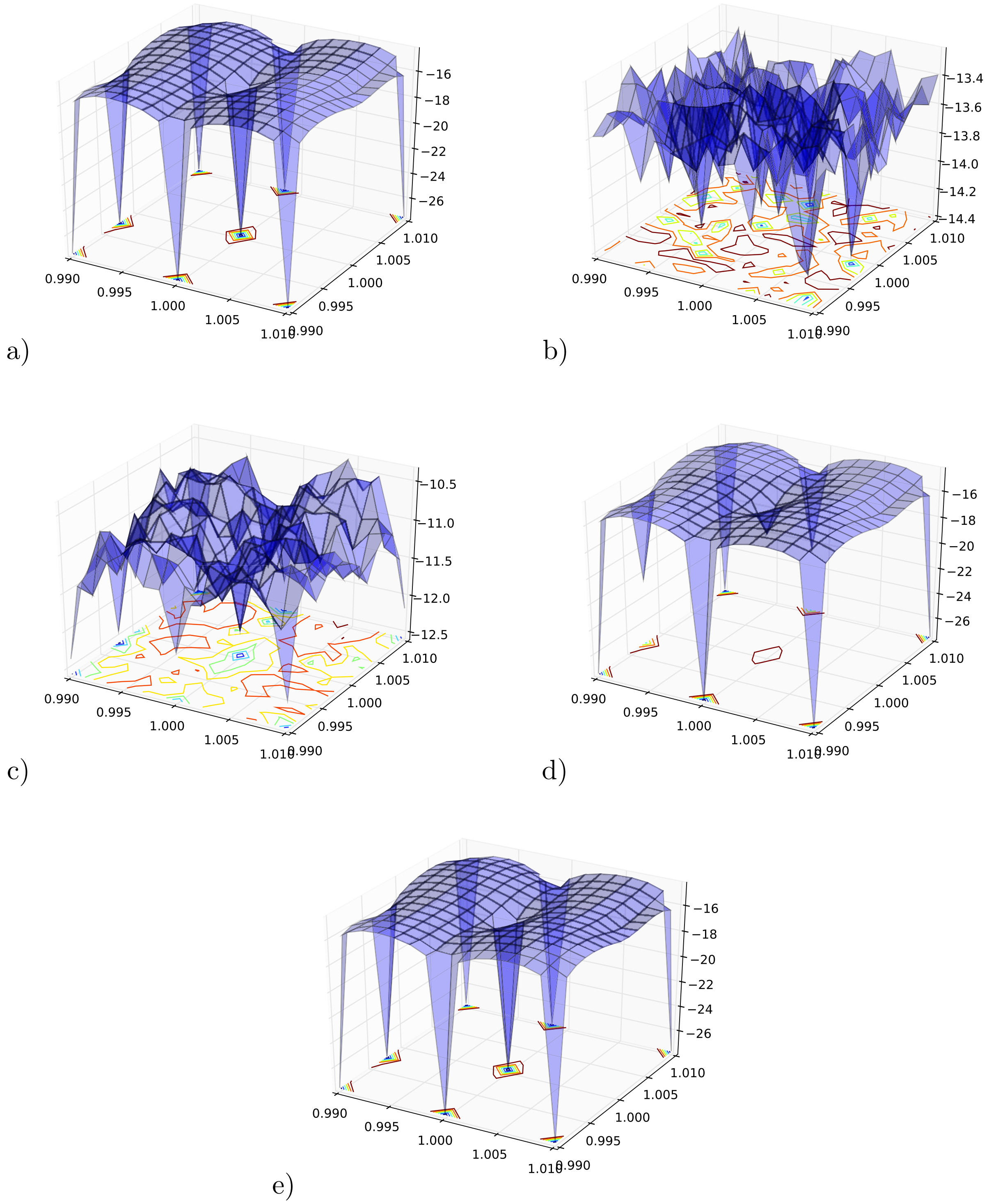

FIGURE 10. Error bound curves (logarithmic scale) as a function of the impedance coefficients: a) $\mathcal{E}_1$, b) $\mathcal{E}_2$, c) $\mathcal{E}_3$, d) $\mathcal{E}_4$, and e) $\mathcal{E}_2$ computed in quadruple precision.

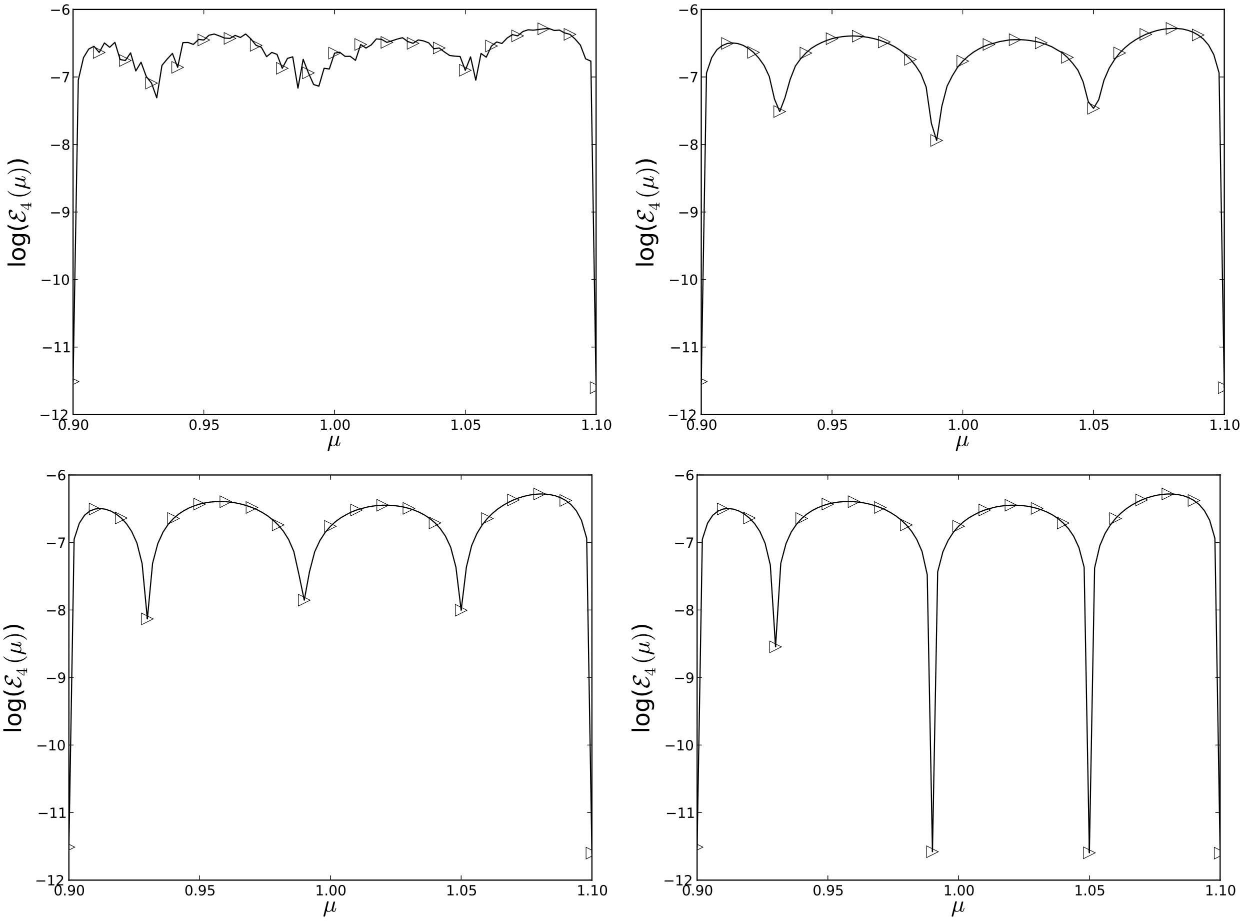

FIGURE 11. Error bound curve for $\mathcal{E}_4$ with respect to the impedance coefficient, with $\hat{N} = 5$ and $\hat{\sigma}$ equal to 14, 30, 40, and 50 (from left to right and top to bottom).

as before, a very good performance. We see in Figure 10 that $\underset{\mu \in \mathcal{P}_{\text{select}}}{\operatorname{argmax}} (\mathcal{E}_4(\mu)) = (1, 1)$ and $\mathcal{E}_4(1, 1) \approx 10^{-16}$; therefore, the formula $\mathcal{E}_4$ with $\hat{\sigma} = 60$ is valid for computing the error bound in Algorithm 1 with tol $= 10^{-16}$.

The behavior of $\mathcal{E}_4$ when $\hat{\sigma}$ increases is investigated in Figure 11 for test case (i). We consider the values $\hat{\sigma} = 14, 30, 40$ and 50. These four values lead to the same local maxima, and increasing $\hat{\sigma}$ allows the formula $\mathcal{E}_4$ to be valid for smaller tolerances (respectively $5 \times 10^{-8}$, $10^{-8}$, $8 \times 10^{-9}$ and $2 \times 10^{-9}$). Another interesting observation comes from considering the fourth plot in Figure 9 and the first plot in Figure 11: the classical formula $\mathcal{E}_2$ requires 16 offline resolutions of (2.12) and stagnates at $10^{-4}$ while the formula $\mathcal{E}_4$ with $\hat{\sigma} = 14$ only requires 14 offline resolutions of (2.12) and is valid for tolerances down to $5 \times 10^{-8}$. This shows that at least in some regimes, the new formula $\mathcal{E}_4$ is valid for lower tolerances than the classical formula $\mathcal{E}_2$, and requires less precomputations. However, contrary to $\mathcal{E}_2$, using $\mathcal{E}_4$ requires that all the quantities $V_r$ defined in (4.2) be recomputed when adding a new vector to the reduced basis.

## CONCLUSION

In this work, we have extended the ideas of [9] by proposing a more stable numerical procedure, using the empirical interpolation method, to represent the *a posteriori* error bound in the reduced basis method as a linear combination of its values at given parameter values, called interpolation points. Moreover, the proposed method provides a way of choosing the interpolation points, and yields better accuracy levels than the classical *a posteriori* error bound and than the procedure proposed in [9]. Besides, our new procedure may require less precomputations than the classical *a posteriori* error bound. The new error bound derived herein can be of particular interest in two situations: (i) when the stability constant of the original problem is very small (this is the case in many practical problems), (ii) when very accurate solutions are needed.

*Acknowledgements.* This work was partially supported by EADS-IW. The authors wish to thank Anthony Patera (MIT) for fruitful discussions.

## References

[1] Z. Bai and D. Skoogh, Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Appl. Numer. Math.* **43** (2002) 9–44.

[2] M.A. Bahayou, Sur le problème de Helmholtz. *Rendiconti del Seminario matematico della Università e Politecnico di Torino* (2007) 427–450.

[3] M. Barrault, Y. Maday, N.C. Nguyen and A.T. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Math. Acad. Sci. Paris* **339** (2004) 667–672.

[4] A. Björck and C.C. Paige, Loss and recapture of orthogonality in the modified Gram–Schmidt algorithm. *SIAM J. Matrix Anal. Appl.* **13** (1992) 176–190.

[5] S. Boyaval, *Mathematical modelling and numerical simulation in materials science.* Ph.D. thesis, Université Paris-Est (2009).

[6] A. Buffa and R. Hiptmair, Regularized combined field integral equations. *Numer. Math.* **100** (2005) 1–19.

[7] R.L. Burden and J.D. Faires, *Numerical Analysis.* PWS Publishing Company (1993).

[8] E. Cancès, V. Ehrlacher and T. Lelièvre, Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. *Math. Models Methods Appl. Sci.* **21** (2011) 2433–2467.

[9] F. Casenave, Accurate *a posteriori* error evaluation in the reduced basis method. *C. R. Math. Acad. Sci. Paris* **350** (2012) 539–542.

[10] F. Casenave, Ph.D. thesis, in preparation (2013).

[11] F. Casenave, M. Ghattassi and R. Joubaud, A multiscale problem in thermal science. *ESAIM: Proceedings* **38** (2012) 202–219.

[12] A. Chatterjee, An introduction to the proper orthogonal decomposition. *Curr. Sci.* **78** (2000) 808–817.

[13] Y. Chen, J.S. Hesthaven, Y. Maday, J. Rodriguez and X. Zhu, Certified reduced basis method for electromagnetic scattering and radar cross section estimation. Technical Report 2011-28, Scientific Computing Group, Brown University, Providence, RI, USA (2011).

[14] Y. Chen, J.S. Hesthaven, Y. Maday and J. Rodríguez, Improved successive constraint method based *a posteriori* error estimate for reduced basis approximation of 2D Maxwell's problem. *ESAIM: M2AN* **43** (2009) 1099–1116.

[15] F. Chinesta, P. Ladeveze and C. Elías, A short review on model order reduction based on proper generalized decomposition. *Arch. Comput. Methods Eng.* **18** (2011) 395–404.

[16] A. Delnevo, I. Terrasse, Code ACTI3S harmonique : Justifications Mathématiques : Partie I. Technical report, EADS CCR (2001).

[17] A. Delnevo, I. Terrasse, Code ACTI3S, Justifications Mathématiques : Partie II, présence d'un écoulement uniforme. Technical report, EADS CCR (2002).

[18] A. Ern and J.L. Guermond, Theory and Practice of Finite Elements, in vol. 159 of *Applied Mathematical Sciences.* Springer (2004).

[19] M. Fares, J.S. Hesthaven, Y. Maday and B. Stamm, The reduced basis method for the electric field integral equation. *J. Comput. Phys.* **230** (2011) 5532–5555.

[20] L. Giraud and J. Langou, When modified Gram–Schmidt generates a well-conditioned set of vectors. *IMA J. Numer. Anal.* **22** (2002) 521–528.

[21] D. Goldberg, What every computer scientist should know about floating point arithmetic. *ACM Computing Surveys* **23** (1991) 5–48.

[22] G.H. Golub and C.F. Van Loan, *Matrix Computations.* Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press (1996).

[23] R.J. Guyan, Reduction of stiffness and mass matrices. *AIAA J.* **3** (1965) 380.

[24] R. Hiptmair, Coercive combined field integral equations. *J. Numer. Math.* **11** (2003) 115–134.

[25] R. Hiptmair and P. Meury, *Stable FEM-BEM Coupling for Helmholtz Transmission Problems.* ETH, Seminar für Angewandte Mathematik (2005).

[26] G.C. Hsiao and W.L. Wendland, *Boundary Element Methods: Foundation and Error Analysis.* John Wiley & Sons, Ltd (2004).

[27] D.B.P. Huynh, G. Rozza, S. Sen and A.T. Patera, A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *C. R. Math. Acad. Sci. Paris* **345** (2007) 473–478.

[28] P. Langlois, S. Graillat and N. Louvet, Compensated Horner scheme. Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2006).

[29] L. Machiels, Y. Maday, I.B. Oliveira, A.T. Patera and D.V. Rovas, Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *C. R. Math. Acad. Sci. Paris* **331** (2000) 153–158.

[30] Y. Maday, N.C. Nguyen, A.T. Patera and S. Pau, A general multipurpose interpolation procedure: the magic points. *Commun. Pure Appl. Anal.* **8** (2008) 383–404.

[31] W.C.H. McLean, *Strongly Elliptic Systems and Boundary Integral Equations.* Cambridge University Press (2000).

[32] A. Nouy and O.P. Le Maître, Generalized spectral decomposition for stochastic nonlinear problems. *J. Comput. Phys.* **228** (2009) 202–235.

[33] A.T. Patera, *Private communication* (2012).

[34] A.T. Patera and G. Rozza, *Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations.* MIT Pappalardo Graduate Monographs in Mechanical Engineering (2007).

[35] M. Paz, Dynamic condensation. *AIAA J.* **22** (1984) 724–727.

[36] C. Prud'homme, D.V. Rovas, K. Veroy, L. Machiels, Y. Maday, A.T. Patera and G. Turinici, Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods. *J. Fluids Eng.* **124** (2002) 70–80.

[37] S.A. Sauter and C. Schwab, *Boundary Element Methods.* Springer Series in Computational Mathematics. Springer (2010).

[38] I.E. Shparlinski, Sparse polynomial approximation in finite fields. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing, STOC '01.* ACM, New York, USA (2001) 209–215.

[39] K. Veroy and A.T. Patera, Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-basis *a posteriori* error bounds. *Int. J. Numer. Methods Fluids* **47** (2005) 773–788.

[40] K. Veroy, C. Prud'homme and A.T. Patera, Reduced-basis approximation of the viscous Burgers equation: rigorous *a posteriori* error bounds. *C. R. Math. Acad. Sci. Paris* **337** (2003) 619–624.