Rigorous and heuristic strategies for the approximation of stability factors in nonlinear parametrized PDEs

Andrea Manzoni, Federico Negri
Rigorous and heuristic strategies for the approximation of stability factors in nonlinear parametrized PDEs

Andrea Manzoni · Federico Negri

Received: date / Accepted: date

Abstract In this paper we present both rigorous and heuristic strategies to compute rapid and reliable lower bounds to stability factors in nonlinear, inf-sup stable parametrized PDEs. Evaluating these quantities efficiently is crucial for the rapid construction of a posteriori error estimates to reduced basis approximations, exploited for both the Offline construction of the reduced space and the Online certification of the reduced solution. First of all, we describe a nonlinear extension of the Successive Constraint Method (SCM) algorithm, based on the computation of local lower bounds and a greedy algorithm for an optimal patching of the parameter space. Because of the large computational costs entailed by the solution of several eigenproblems (and linearization of nonlinear operators), we turn to some cheaper heuristic strategies, based on either the computation of piecewise constant approximation or a suitable radial basis interpolant to the stability factor. Even though they are not always rigorous, heuristic strategies provide very good results but entailing much lower computational costs. We illustrate the efficacy of these procedures through some numerical test cases dealing with parametrized Navier-Stokes equations.

Keywords Stability factors · nonlinear parametrized PDEs · reduced basis methods · Brezzi-Rappaz-Raviart theory

Mathematics Subject Classification (2000) 65N12 · 65N30 · 76D05 · 78M34

A. Manzoni
SISSA Mathlab, International School for Advanced Studies,
Via Bonomea 265, I-34136 Trieste, Italy
E-mail: amanzoni@sissa.it

F. Negri
CMCS-MATHICSE-SB, Ecole Polytechnique Fédérale de Lausanne,
Station 8, CH-1015 Lausanne, Switzerland
E-mail: federico.negri@epfl.ch
1 Introduction

Stability factors of differential operators are relevant for the well-posedness analysis of problems governed by PDEs and enter in the error estimates of any numerical approximation method. In fact, the inverse of the stability factor appears in both stability estimates and (a posteriori) error estimates. Their rapid and reliable evaluation is crucial especially when dealing with:

- parametrized PDEs, i.e. when the PDEs at hand depend on a vector of $p \geq 1$ input parameters $\mu \in D \subset \mathbb{R}^p$, which may be related either physical properties or geometrical features;
- nonlinear PDEs, where the nonlinear behavior may give rise to complicated dynamics and, possibly, bifurcation phenomena, which reflect on strong changes of the solution and, also on the difficulty to keep the effect of numerical errors on the data under control.

In this paper we focus on nonlinear parametrized PDEs, which feature both the issues above; as a motivation, we mention the steady Navier-Stokes equations in a backward facing step domain, parametrized with respect to the Reynolds number and/or the domain aspect ratio. In particular, we aim at computing – in a very efficient way – a numerical approximation of the solution $u(\mu)$, by varying the parameter values $\mu$ in a suitable parameter space $D$. This matters in all cases where many input/field/output evaluations are required, such as in optimization or sensitivity analyses.

To do this, we rely on the Reduced Basis (RB) method, which allows to compute a reduced approximation $u_N(\mu) \in V_N$ of the PDE solution $u(\mu) \in V$, for any $\mu \in D$, as a linear combination of snapshots $\{u_h(\mu_1), \ldots, u_h(\mu_N)\}$ corresponding to a small set of sampled parameter values $\mu_1, \ldots, \mu_N$, through a Galerkin projection in the low-dimensional subspace $V_N = \text{span}\{u_h(\mu_1), \ldots, u_h(\mu_N)\}$. Here $V$ is a suitable Hilbert space, $V_h$ is a high-fidelity approximation space of dimension $N_h$ and $u_h(\mu) \in V_h$ is the high-fidelity approximation to $u(\mu)$ obtained by any kind of numerical discretization. Not only, we aim at providing a rigorous a posteriori error bound, which usually takes the form

$$\|u_h(\mu) - u_N(\mu)\|_V \leq \frac{1}{\beta_h(u_N(\mu))} \Phi(\|r(u_N(\mu))\|_{V'})$$

where $r(u_N(\mu))$ is the residual of the reduced approximation, $\|r(u_N(\mu))\|_{V'}$ its dual norm, $\Phi : \mathbb{R} \to \mathbb{R}^+$ a suitable real function and $\beta_h(u_N(\mu))$ a stability factor, related with the (discrete, high-fidelity approximation of the) differential operator, to be bounded from below.

Furthermore, $\beta_h$ and $\beta_N$ are called stability factors because they affect the stability estimates

$$\|\tilde{u}_h(\mu)\|_V \leq \frac{1}{\beta_h(u_h(\mu))} \|\tilde{f}\|_{V'}, \quad \|\tilde{u}_N(\mu)\|_V \leq \frac{1}{\beta_N(u_N(\mu))} \|\tilde{f}\|_{V'},$$

for the perturbations $\tilde{u}_h(\mu)$, $\tilde{u}_N(\mu)$ in both the high-fidelity approximation $u_h(\mu)$ and the reduced approximation $u_N(\mu)$, in response to a given disturbance $\tilde{f} \in V'$. Here we denote by $\beta_N(u_N(\mu))$ the stability factor related with the discrete, reduced operator.
We emphasize that $\beta_h(u_N(\mu))$ can be computed by solving, for any $\mu \in \mathcal{D}$, a generalized eigenproblem of dimension $N_h$; in the nonlinear case this latter still depends on the reduced approximation $u_N(\mu)$ (see Sect. 3). Since the stability factor $\beta_h(\mu)$ – the one related to the high-fidelity discretization – both affects the error bound and (under suitable assumptions, and by including suitable correction factors) can be related to the stability of the reduced approximation, too, evaluating tight lower bounds of this quantity is extremely relevant.

With this aim, the so-called Successive Constraint Method has been developed for the special requirements of the RB method, such as an efficient Offline-Online strategy [21]. SCM is an iterative procedure based on the successive solution of suitable linear optimization problems, which has been introduced in [14]; see also [6,7]. A general version using the so-called natural norm [26] has been analyzed in [12]; a recent application to Stokes equations is shown in [24]. Despite of its generality, SCM very often suffers of slow convergence, already in the linear case, when dealing with $p \geq 3$ parameters (see e.g. [12,18]), thus calling into play the need of more efficient (yet accurate) procedures.

The goal of this paper is twofold:

1. first, we review (SCM) to compute lower bounds of stability factors related to parametrized, inf-sup stable, nonlinear operators. Here we show some theoretical results, which allow to apply SCM to nonlinear operators. Despite some ideas were anticipated in [18] and, more recently, in [29,30], a rigorous justification is provided in this paper;
2. then, in order to overcome some computational bottlenecks entailed by SCM, we propose some alternative, heuristic strategies, based either on minimization or (adaptive) interpolation procedures. We show a numerical comparison between these approaches, in order to assess their effectiveness.

In more details, we characterize in this paper some theoretical results dealing with the Lipschitz-continuity of affine, parametrized nonlinear operators, in order to provide a rigorous extension of the (linear version of) SCM to nonlinear operators. In particular, we develop a SCM procedure which relies on a $\mu$-independent notion of norm and which can be performed before (and independently from) the generation of a reduced approximation, for the sake of computational savings. Even if the Online stage, i.e. each evaluation $\mu \rightarrow \beta_h(u_N(\mu))$, can be performed very rapidly – this is mandatory in view of an efficient construction and certification of the reduced approximation, see [21,25] – and without computing the reduced approximation $u_N(\mu)$ to the PDE solution, a very expensive Offline stage required to construct local lower bounds to $\beta_h(u_N(\mu))$ in the parameter space might compromise the efficiency of the whole reduction process. In fact, the slow convergence of SCM is even more evident in the nonlinear case, as shown by the numerical test cases discussed in Sect. 6.

For these reasons, we propose some heuristic strategies to compute a lower bound to the stability factor of parametrized, inf-sup stable, nonlinear operators. These strategies may require either the computation of piecewise constant approximations of the stability factor, or the construction of suitable radial basis interpolants, through an adaptive choice of the interpolation points in the parameter space. In this way, it is possible to obtain reliable lower bounds, whose Offline construction and Online evaluation prove to be extremely faster.
The structure of the paper is as follows. In Sect. 2 we provide some background notions related with stability factors for nonlinear, inf-sup stable parametrized PDEs, and some theoretical results required to extend the SCM procedure to this case. In Sect. 3 both the high-fidelity and the reduced approximation of this class of problems are introduced. In Sect. 4 we present the extension of the SCM algorithm to the case of nonlinear inf-sup stable operators. Then, in Sect. 5 we present some possible heuristic strategies to circumvent the computational burden involved in the nonlinear SCM algorithm. In the end, we show in Sect. 6 some numerical test cases dealing with steady parametrized Navier-Stokes equations in order to compare their computational performances.

2 Stability factors for nonlinear, inf-sup stable parametrized PDEs

Given a regular spatial domain \( \Omega \subset \mathbb{R}^d \), a bounded closed parameter domain \( \mathcal{D} \subset \mathbb{R}^p \) whose elements are denoted by \( \mu \in \mathcal{D} \), let us indicate by \( V = V(\Omega) \) a Hilbert space with inner product \((\cdot, \cdot)_V\) and induced norm \( \| \cdot \|_V = \sqrt{(\cdot, \cdot)_V} \).

We focus on the case of stationary, quadratically nonlinear parametrized operators, for which our problem of interest can be expressed as follows: given \( \mu \in \mathcal{D} \), find \( u = u(\mu) \in V \) s.t.

\[
A(u(\mu), v; \mu) = a(u(\mu), v; \mu) + c(u(\mu), v; \mu) = f(v; \mu) \quad \forall \ v \in V, \tag{2}
\]

where \( a(\cdot, \cdot; \mu) \) is a continuous, inf-sup stable bilinear form over \( V \times V \) and \( c(\cdot, \cdot, \cdot; \mu) \) is a continuous trilinear form over \( V \times V \times V \). Here the right-hand side is a (possibly) parametrized linear form \( f(\cdot; \mu) : V \to \mathbb{R} \), given by

\[
f(v; \mu) = \langle v, F(\mu) \rangle_V,
\]

being \( F(\mu) \in V' \) and \( V' = L(V; \mathbb{R}) \) the dual space of \( V \).

According to the general Brezzi-Rappaz-Raviart (BRR) theory \cite{2}, problem (2) is well posed if and only if the following continuity and (Babuška) inf-sup conditions hold:

\[
\gamma(\mu) = \sup_{v \in V} \sup_{w \in V} \frac{dA(u(\mu); \mu)(v, w)}{\|v\|_V \|w\|_V} < +\infty, \quad \forall \ \mu \in \mathcal{D}, \tag{3}
\]

\[
\exists \ \beta(\mu) > 0 : \ \beta(\mu) = \inf_{v \in V} \sup_{w \in V} \frac{dA(u(\mu); \mu)(v, w)}{\|v\|_V \|w\|_V} \geq \beta(\mu), \quad \forall \ \mu \in \mathcal{D}. \tag{4}
\]

In fact, these conditions ensure the existence of a local branch of non-singular solutions \cite{9}, see Proposition 1. Here \( dA(u(\mu); \mu)(\cdot, \cdot) \) denotes the Fréchet derivative of \( A(\cdot, \cdot; \mu) \) with respect to the first variable, which is given, at \( z \in V \), by

\[
dA(z; \mu)(w, v) = a(w, v; \mu) + c(z, w, v; \mu) + c(w, z, v; \mu) \quad \forall \ w, z, v \in V. \tag{5}
\]

From now on, we denote \( d(z; \mu)(w, v) = c(z, w, v; \mu) + c(w, z, v; \mu) \). Furthermore,

\[
\gamma^a(\mu) = \sup_{v \in V} \sup_{w \in V} \frac{a(v, w; \mu)}{\|v\|_V \|w\|_V} < +\infty, \quad \gamma^\gamma(\mu) = \sup_{u \in V} \sup_{v \in V} \sup_{w \in V} \frac{c(u, v, w; \mu)}{\|u\|_V \|v\|_V \|w\|_V} < +\infty
\]
denote the (μ-dependent) continuity constants of \(a(\cdot, \cdot; \mu)\) and \(c(\cdot, \cdot, \cdot; \mu)\), respectively, while
\[
\gamma_d^d(\mu) = \sup_{u \in V} \sup_{w \in V} \frac{d(u(\mu); \mu)(v, w)}{\|v\|_V\|w\|_V} < +\infty
\]
denotes the continuity constant of \(d(u; \mu)(\cdot, \cdot)\).

We point out that, even if the stability factor \(\beta(\mu)\) we want to bound from
below is related with the left-hand side of (2), in the nonlinear case it depends on
the solution \(u(\mu)\), and thus also on the right-hand side \(f(\cdot; \mu)\) of (2), indirectly.
This is a further difficulty which makes the evaluation of a good lower bound to
\(\beta(\mu)\) much more involved in the nonlinear case.

The goal of this work is to provide some efficient numerical strategies (both
rigorous and heuristic) to estimate a parametric lower bound to the stability factor
\(\beta(\mu)\) defined by (4). To frame this problem in a suitable setting – from both a
functional and computational standpoint – we need to introduce some definitions
and provide some basic results concerning the continuity and the regularity of the
map \(\mu \mapsto u(\mu)\); this is the goal of the following subsections.

2.1 Supremizer operator, norms and parametric dependence

We introduce the parametrized linear operator \(T^\mu : V \to V\) such that, for any
\(\mu \in \mathcal{D}, v \in V\),
\[
(T^\mu v, w)_V = dA(u(\mu); \mu)(v, w) \quad \forall w \in V;
\]
equivalently, by Riesz theorem,
\[
T^\mu v = \arg \sup_{w \in V} \frac{dA(u(\mu); \mu)(v, w)}{\|w\|_V}, \quad \forall v \in V.
\]
Because of (7), \(T^\mu\) is called supremizer operator. It follows that (3) and (4) can
be equivalently expressed as
\[
\gamma(\mu) = \sup_{w \in V} \frac{\|T^\mu w\|_V}{\|w\|_V}, \quad \beta(\mu) = \inf_{w \in V} \frac{\|T^\mu w\|_V}{\|w\|_V}.
\]
Assuming that \(0 < \beta^0(\mu) \leq \beta(\mu)\) and \(\gamma(\mu) < \infty\) for each \(\mu \in \mathcal{D}\), implies that
\[
\|||w|||_\mu := \|T^\mu w\|_V \quad \forall w \in V,
\]
defines a norm, usually referred to [26,8] as natural norm. Thanks to (8), it is
equivalent to the V-norm,
\[
\frac{1}{\gamma(\mu)}\|T^\mu w\|_V \leq \|w\|_V \leq \frac{1}{\beta(\mu)}\|T^\mu w\|_V.
\]
In order to develop an Offline/Online strategy, we assume that the forms
appearing in (2) fulfill the following parameter separability – also called affine pa-
rameter dependence – property:
\[
a(u, v; \mu) = \sum_{q=1}^{Q_\alpha} \Theta^a_q(\mu) a_q(u, v), \quad c(u, v, w; \mu) = \sum_{q=1}^{Q_\alpha} \Theta^c_q(\mu) c_q(u, v, w) \quad \forall u, v, w \in V,
\]
(11)
for some integers $Q_a, Q_c$, where $\Theta_q^0, \Theta_q^c \in C^1(D)$ and $a_q(\cdot, \cdot), c_q(\cdot, \cdot, \cdot)$ are continuous bilinear (trilinear) forms over $V \times V$ ($V \times V \times V$), respectively; moreover, we set $Q_A = Q_a + Q_c$. The requirement that $\Theta_q^0, \Theta_q^c$ are of class $C^1(D)$ is essential to ensure that the map $\mu \mapsto u(\mu)$ is regular enough, as we will see in the following subsection.

As before, we denote the (now $\mu$-independent) continuity constants of $a_q(\cdot, \cdot)$ and $c_q(\cdot, \cdot, \cdot)$ by

$$\gamma_q^d = \sup_{v \in V} \sup_{w \in V} \frac{a_q(v, w)}{\|v\|V \|w\|V} < +\infty, \quad \gamma_q^c = \sup_{u \in V} \sup_{v \in V} \sup_{w \in V} \frac{c_q(u, v, w)}{\|u\|V \|v\|V \|w\|V} < +\infty,$$

respectively. In the same way, we set

$$d(z; \mu)(w, v) = \sum_{q=1}^{Q_a} \Theta_q^0(\mu)(c_q(z, w, v) + c_q(w, z, v)) = \sum_{q=1}^{Q_a} \Theta_q^c(\mu)d_q(z)(v, w),$$

and denote by

$$\gamma_q^d(u) = \sup_{v \in V} \sup_{w \in V} \frac{d_q(u)(v, w)}{\|v\|V \|w\|V} < +\infty,$$

the ($u$-dependent) continuity constants of $d_q(u)(\cdot, \cdot)$. All these quantities play a fundamental role in the nonlinear SCM algorithm, as we will see in Sect. 4.

2.2 Fréchet derivatives of operators and regularity of solutions

Let us show some theoretical results required to ensure the well-posedness of the nonlinear extension of the SCM procedure. For the sake of generality, let us rewrite problem (2) under the form

$$\mathcal{E}(\mu, u) = 0 \quad \text{in } V',$$

being $\mathcal{E} : D \subset \mathbb{R}^p \times V \to V'$ the operator defined as follows:

$$V' \langle \mathcal{E}(\mu, z), w \rangle_V = A(z, w; \mu) - f(w; \mu).$$

Moreover, let us denote by $d_a \mathcal{E} : D \times V \to V$ and $d_\mu \mathcal{E} : D \subset \mathbb{R}^p \times V \to \mathbb{R}^p$ the (partial) Fréchet derivatives of $\mathcal{E}$; note that we can identify $\mathbb{R}^p$ and its dual.

**Proposition 1** For the parametrized operator $A(\cdot, \cdot; \mu) : V \times V \to \mathbb{R}$ defined in (2), suppose that:

1. the continuity and the inf-sup conditions (3)–(4) hold;
2. the parameter separability assumption (11) holds, being $\Theta_q^0, \Theta_q^c : D \to \mathbb{R}$, $q = 1, \ldots, Q_a$, $q' = 1, \ldots, Q_c$, prescribed $C^1$ functions.

Then, if for some $\mu_0 \in D$, $u_0 \in V$, $\mathcal{E}(\mu_0, u_0) = 0$, there exist $r_0, r > 0$ such that, for every $\mu \in B_{r_0}(\mu_0) \cap D$, there exists a unique $u(\mu) \in B_r(u_0) \cap V$ such that $\mathcal{E}(\mu, u(\mu)) = 0$. Furthermore, the map $\mu \mapsto u(\mu)$ is Lipschitz continuous and

$$u'(\mu) = - (d_a \mathcal{E}(\mu, u(\mu)))^{-1} d_\mu \mathcal{E}(\mu, u(\mu)).$$
Proof The proof is a direct consequence of the Implicit Function Theorem: we refer here to the version stated by Hildebrandt and Graves [11], which can also be found, e.g., in [31]. A very general version providing further insights on the Lipschitz constant of the map $\mu \mapsto u(\mu)$ can be found, e.g., in [4,15], to which we refer the reader.

Provided that the continuity condition (3) holds, $d_uE$ is continuous at each point $(\mu_0, u_0) \in D \times V$; its inverse is a continuous linear operator thanks to the inf-sup condition (4) – in other words, $d_uE$ is an isomorphism, for any $(\mu_0, u_0) \in D \times V$. Furthermore, if the parameter separability assumption (11) holds, being given $C^1$ functions $\Theta^o_q, \Theta^c_q : D \to \mathbb{R}$, $q = 1, \ldots, Q_a$, $q' = 1, \ldots, Q_c$, $E$ is a $C^1$ map. Then, the Implicit Function Theorem provides the existence of $r_0, r > 0$ and of a (unique) $C^1$ map $\mu \mapsto u(\mu)$ such that, for every $\mu \in B_{r_0}(\mu_0) \cap D$, $E(\mu, u(\mu)) = 0$. □

Proposition 2 Under the assumptions of Proposition 1, and by assuming that

$$\| u(\mu) \|_V \leq K_u, \quad \forall \mu \in D,$$

there exists a positive constant $C > 0$ such that, for any $\mu, \mu^\star \in D$, $\mu \neq \mu^\star$,

$$\| dA(u(\mu); \mu)(v, w) - dA(u(\mu^\star); \mu^\star)(v, w) \| \leq C |\mu - \mu^\star| \| v \|_V \| w \|_V. \quad (16)$$

Furthermore, the following estimate holds:

$$\| T^\mu w - T^{\mu^\star} w \|_V \leq \frac{C}{\gamma(\mu^\star)} |\mu - \mu^\star| \| T^{\mu^\star} w \|_V \quad \forall w \in V. \quad (17)$$

The proof is reported in Appendix A.1. This result will be exploited in Sect. 4.1 to justify the Offline/Online SCM procedure in the nonlinear case. As a matter of fact, a global lower bound to the stability factor can be evaluated by solving a sequence of local problems, which result to be coercive thanks to property (16).

Remark 1 In the Navier-Stokes case, an a priori estimate like (15) follows the coercivity of the bilinear form $a(\cdot, \cdot; \mu)$ and the skew-symmetry (with respect to the last two arguments) of the trilinear form $c(\cdot, \cdot, \cdot; \mu)$; see e.g. [27], Sect. 2.1.

3 High-fidelity and reduced approximation

In this section we introduce the high-fidelity approximation of problem (2), based on a Galerkin-Finite Element (FE) method, and then that based on the RB method. Moreover, we discuss some stability issues related with these two approximation strategies, as well as a general a posteriori error estimate, where the role of stability factors is highlighted.

Let us denote by $V_h \subset V$ a FE approximation space of dimension $N_h$, with inherited inner product $(v, w)_{V_h} = (v, w)_V$ and norm $\| v \|_{V_h} = \| v \|_V$. The Galerkin-FE approximation of (2) reads as follows: given $\mu \in D$, find $u_h(\mu) \in V_h$ s.t.

$$A(u_h(\mu), v_h; \mu) = f(v_h; \mu) \quad \forall v_h \in V_h. \quad (18)$$

Problem (18) is equivalent to the following algebraic nonlinear system:

$$(K(\mu) + C(u_h(\mu); \mu)) u_h(\mu) = f(\mu), \quad (19)$$
being $u_h(\mu) \in \mathbb{R}^{N_h}$ the vector representation of $u_h(\mu) \in V_h$ over a Lagrangian basis $\{\varphi_j^h\}_{j=1}^{N_h}$ of $V_h$ – i.e. such that $\varphi_j^h(x) = v_h(x_i)$ for any $v_h \in V_h$ – and

$$(\mathbb{R}(\mu))_{ij} = a(\varphi_i^h, \varphi_j^h; \mu), \quad (C(w_h; \mu))_{ij} = c(w_h, \varphi_i^h, \varphi_j^h; \mu), \quad f^{(ij)}(\mu) = f(\varphi_i^h; \mu),$$

the matrices corresponding to the linear and the nonlinear term, and the vector corresponding to the source term, respectively $(i, j = 1, \ldots, N_h)$.

Concerning the stability of the approximation (18), we rely on the Brezzi-Rappaz-Raviart theory [2, 4]. As for the original problem, let us assume that:

$$\gamma_h(\mu) = \sup_{v_h \in V_h} \sup_{w_h \in V_h} \frac{dA(u_h(\mu); \mu)(v_h, w_h)}{\|v_h\|_V \|w_h\|_V} < +\infty, \quad \forall \mu \in D, \quad (20)$$

$$\exists \beta_0(\mu) > 0 : \beta_h(\mu) = \inf_{v_h \in V_h, w_h \in V_h} \frac{dA(u_h(\mu); \mu)(v_h, w_h)}{\|v_h\|_V \|w_h\|_V} \geq \beta_0(\mu), \quad \forall \mu \in D. \quad (21)$$

We underline that, in the definition above, $\beta_h(\mu) = \beta_h(u_h(\mu))$: this will be particularly relevant when dealing with the reduced solution $u_N(\mu)$, but we omit the dependence on the solution, as in (21), wherever it is clear from the context. Then, if $V_h$ is chosen accordingly to these conditions – which are, in fact, the discrete version of (3)–(4) – problem (18) admits a unique solution.

Concerning the regularity of the solution with respect to $\mu$, a result similar to that of Proposition 1 can be proved if we consider a Galerkin approximation, i.e. find $u_h \in V_h$ s.t.

$$V^c(\mathcal{E}_h(\mu), u_h) = 0 \quad \forall u_h \in V_h, \quad (22)$$

being $\mathcal{E}_h : \mathcal{D} \times V \rightarrow V'$ a suitable\(^1\) approximation of the operator defined by (14). In particular, if $\mathcal{E}_h$ is a $C^m$-map, then $\mu \mapsto u_h(\mu)$ is also of class $C^m$ (see e.g. [4, Chapter 12 and Remark 13.2]).

By introducing the discrete supremizer operator\(^2\) $T^\mu : V_h \rightarrow V_h$ s.t.

$$(T^\mu v_h, w_h)_V = dA(u_h(\mu); \mu)(v_h, w_h) \quad \forall w_h \in V_h, \quad (23)$$

we have that

$$(\beta_h(u_h(\mu)))^2 = \left( \inf_{v \in V_h} \frac{dA(u_h(\mu); \mu)(v, T^\mu v)}{\|v\|_V \|T^\mu v\|_V} \right)^2 \geq \inf_{v \in V_h} \frac{\|T^\mu v\|_V^2}{\|v\|_V^2}. \quad (24)$$

The algebraic counterpart of (24) can be obtained by introducing the matrix norm

$$\mathcal{X}_{ij} = (\varphi_i^h, \varphi_j^h)_V, \quad i, j = 1, \ldots, N_h \quad (25)$$

of $V_h$, so that $\|v_h\|_V^2 = \mathcal{X} v_h^T v_h$ for any $v_h \in V_h$. Moreover, denoting by $t_h$ the vector of components $t_h^{(i)} = (T^\mu v_h)(x_i)$, we have that

$$w_h^T \mathcal{X} t_h = w_h^T \mathcal{X} v_h, \quad (26)$$

\(^1\) Lipschitz continuity of the Fréchet derivative of $\mathcal{E}_h$, as well as consistency and stability conditions, must be ensured. For instance, Taylor-Hood elements [9] allow to meet these requirements in the Navier-Stokes case.

\(^2\) For the sake of notation, we denote by $T^\mu$ the discrete supremizer operator, too.
from (24) and (26) we find

detailed review of RB methods.

a posteriori
which enters in the following
through a Galerkin

Rigorous and heuristic strategies for the approximation of stability factors 9

"rigid" eigenvalue of
(for details see e.g. [19]).

Rather, we focus here on the estimate of a lower bound

so that \( \beta_h(u_h(\mu)) = (\lambda_{\text{min}}(\mu))^{1/2} \), where \( \lambda_{\text{min}}(\mu) \) is the smallest (generalized) eigenvalue of

\[
F^T(\mu)X^{-1}F(\mu)v_h = \lambda(\mu)Xv_h \quad \forall \ v_h \neq 0.
\]

Thus, it is clear that evaluating the stability factor \( \beta_h(u_h(\mu)) \), for any \( \mu \in \mathcal{D} \), entails the solution of both (19) and (27). For this reason, finding reliable and rapid (i.e., independent of \( N_h \)) ways to obtain tight lower bounds of \( \beta_h(u_h(\mu)) \) is a key factor in order to improve any procedure exploiting the evaluation of this quantity, such as the RB method recalled in the following subsection.

3.1 Reduced basis approximation

Our final goal is to compute, for any \( \mu \in \mathcal{D} \), a RB approximation \( u_N(\mu) \in \tilde{V}_N \) to \( u_h(\mu) \), where

\[
\tilde{V}_N = \text{span}\{u_h(\mu^1), \ldots, u_h(\mu^N)\} \subset V_h
\]

is a reduced space, made by \( N \ll N_h \) solutions to problem (18) computed for properly chosen parameter values \( \mu^1, \ldots, \mu^N \). Then, we perform the Gram-Schmidt procedure on the snapshots to obtain an orthonormal basis \( \{\varphi_1, \ldots, \varphi_N\} \), so that we have \( u_N(\mu) = \sum_{j=1}^{N} u_N^{(j)}(\mu)\varphi_j \), where the components \( u_N^{(j)} \) are computed through a Galerkin\(^3 \) projection of (2) over \( \tilde{V}_N \): find \( u_N(\mu) \in \tilde{V}_N \) s.t.

\[
A(u_N(\mu), v_N; \mu) = f(v_N; \mu) \quad \forall \ v_N \in \tilde{V}_N.
\]

Ensuring the stability of the reduced approximation (29) might be troublesome, since it is not automatically fulfilled by defining \( \tilde{V}_N \) as in (28). For this reason, we may enrich \( \tilde{V}_N \) by supremizer functions, and then define \( V_N \) of dimension \( \dim(V_N) = 2N \), in order to enforce the inf-sup condition also at the reduced level (for details see e.g. [19]). Rather, we focus here on the estimate of a lower bound \( \beta_h^{\text{LB}}(u_N(\mu)) \) for the discrete inf-sup stability factor

\[
\beta_h(u_N(\mu)) = \inf_{v \in V_h} \sup_{w \in V_h} \frac{dA(u_N(\mu); \mu)(v, w)}{\|v\|_V \|w\|_V} \quad \forall \mu \in \mathcal{D},
\]

which enters in the following a posteriori error estimate: for any \( N \geq N^*(\mu) \),

\[
\|u_h(\mu) - u_N(\mu)\|_V \leq \frac{\beta_h^{\text{LB}}(u_N(\mu))}{2\gamma_h(\mu)} \left( 1 - \sqrt{1 - \tau_N(\mu)} \right) =: \Delta_N(\mu) \quad \forall \mu \in \mathcal{D},
\]

\(^3\) For the sake of simplicity, here we restrict ourselves to the case of Galerkin projection, although sometimes a more general Petrov-Galerkin method is used. See e.g. [21, 25] for a detailed review of RB methods.
being $\gamma_h^N(\mu)$ the discrete continuity constant of $c(\cdot, \cdot; \mu)$,

$$
\tau_N(\mu) = \frac{4\gamma_h^N(\mu)||r(\cdot; \mu)||_{V'}}{\beta_{LB}^h(u_N(\mu))^2},
$$

$||r(\cdot; \mu)||_{V'} = \sup_{w \in V_N} r(w; \mu)/\|w\|_V$ the dual norm of the residual and $N^*(\mu)$ the smallest $N$ such that $\tau_N(\mu) < 1$, for all $N \geq N^*(\mu)$. See e.g. [28,8,19] for further details and proofs in the Navier-Stokes case, or [5,23] for recent applications to nonlinear advection/diffusion problems.

We highlight that, since the error bound is related to the RB solution $u_N(\mu)$, the stability factor appearing in (31) has to be evaluated with respect to $u_N(\mu)$. However, this is infeasible if we want to compute a parametric lower bound for the stability factor before assembling the reduced space, unless these two procedures are run simultaneously (as shown e.g. in [29]), thus yielding a more expensive offline stage. Thus, our goal is to provide a lower bound $\beta_{LB}^h(u_h(\mu))$ to the stability factor $\beta_h(u_h(\mu))$ evaluated w.r.t. the truth FE solution $u_h(\mu)$

$$
\beta_h(u_h(\mu)) = \inf_{v \in V_h} \sup_{w \in V_h} \frac{dA(u_h(\mu); \mu)(v, w)}{\|v\|_V \|w\|_V}, \quad \forall \mu \in D, \quad (32)
$$

instead of a lower bound $\beta_{LB}^h(u_N(\mu))$ to the stability factor $\beta_h(u_N(\mu))$ defined in (30). In fact, the former quantity provides an asymptotically good approximation to the latter, thanks to the

**Proposition 3** The following relation holds:

$$
|\beta_h(u_h(\mu)) - \beta_h(u_N(\mu))| \leq 2\gamma_h^N(\mu)||u_h(\mu) - u_N(\mu)||_V \quad \forall \mu \in D. \quad (33)
$$

**Proof** By exploiting the trilinearity and the continuity of $c(\cdot, \cdot; \mu)$, we have

$$
\beta_h(u_1) = \inf_{v \in V_h} \sup_{w \in V_h} \left( \frac{a(v, w; \mu) + c(u_2, v, w; \mu) + c(v, u_2, w; \mu)}{\|v\|_V \|w\|_V} + \frac{c(u_1 - u_2, v, w; \mu) + c(v, u_1 - u_2, w; \mu)}{\|v\|_V \|w\|_V} \right)
= \inf_{v \in V_h} \sup_{w \in V_h} \frac{dA(u_2)(v, w) + c(u_1 - u_2, v, w) + c(v, u_1 - u_2, w)}{\|v\|_V \|w\|_V}
\leq 2\gamma_h^N(\mu)||u_1 - u_2||_V = \beta_h(u_2) + 2\gamma_h^N(\mu)||u_1 - u_2||_V.
$$

By considering in the previous inequality first $u_1 = u_N(\mu)$, $u_2 = u_h(\mu)$, and then $u_2 = u_N(\mu)$, $u_1 = u_h(\mu)$, (33) easily follows. \qed

Thus, provided the RB approximation $u_N(\mu)$ is sufficiently close to the high-fidelity approximation $u_h(\mu)$, the discrete stability factor $\beta_h(u_N(\mu))$ related to the former can be properly approximated by the stability factor $\beta_h(u_h(\mu))$ related to the latter, thus yielding the chance to estimate a lower bound to the (discrete) stability factor before assembling the reduced space. This can be ensured, for instance, by choosing a suitable stopping criterion in the greedy procedure often exploited for the offline construction of the reduced space (see e.g. [21,25].)
4 A generalized SCM for estimating a lower bound to $\beta_h(u_h(\mu))$

We now extend the Successive Constraint Method (SCM) [12] to compute a lower bound to the discrete stability factor (32), which can be seen as the solution of a suitable eigenproblem. Following [26, 12], we adopt a natural norm SCM procedure based on the patching of some local inf-sup stability factors properly computed for a set of $J$ parameter values $\mathcal{S} = \{\mu^{(1)}, \ldots, \mu^{(J)}\}$, selected through a greedy procedure. The key observation is provided by the following relation:

$$
\beta_h(u_h(\mu)) = \inf_{v \in V_h} \sup_{w \in V_h} dA(u_h(\mu); \mu; v, w) \frac{||T^* w||_V}{||w||_V}
$$

$$
\geq \inf_{v \in V_h} \sup_{w \in V_h} \frac{dA(u_h(\mu); \mu; v, w)}{||T^* w||_V ||v||_V} \inf_{w \in V_h} ||T^* w||_V
$$

$$
= \beta_{\mu^*}(\mu) \beta_h(u_h(\mu^*)) \geq \tilde{\beta}_{\mu^*}(\mu) \beta_h(u_h(\mu^*))
$$

where $\beta_{\mu^*}(\mu)$ is a surrogate inf-sup stability factor upon $\mu^* \in \mathcal{S}$ and

$$
\tilde{\beta}_{\mu^*}(\mu) := \inf_{v \in V_h} \frac{dA(u_h(\mu); \mu; v, T^* v)}{||T^* v||_V^2}
$$

is a lower bound of $\beta_{\mu^*}(\mu)$ s.t.

$$
\tilde{\beta}_{\mu^*}(\mu) \leq \inf_{v \in V_h} \frac{||T^* v||_V}{||T^* v||_V} = \inf_{v \in V_h} \sup_{w \in V_h} \frac{dA(u_h(\mu); \mu; v, w)}{||T^* v||_V ||w||_V} =: \beta_{\mu^*}(\mu),
$$

thanks to Cauchy-Schwarz inequality and the definition of supremizer operator. As already remarked, $||T^* \cdot||_V$ is an equivalent norm to $||\cdot||_V$ in a neighborhood of $\mu^*$, thanks to (21). However, to be able to evaluate a lower bound of stability factors independently of the construction of a reduced approximation, we rather use the standard $V$-norm wherever possible. We remark that $\beta_{\mu^*}(\mu) = (\lambda_{\mu^*}(\mu))^{1/2}$, where $\lambda_{\mu^*}(\mu)$ is the smallest (generalized) eigenvalue of

$$
\mathbb{F}(\mu^*)^T X^{-1} \mathbb{F}(\mu) v = \lambda_{\mu^*}(\mu) \mathbb{F}(\mu^*)^T X^{-1} \mathbb{F}(\mu^*) v \quad \forall \ v \neq 0.
$$

Thus, in order to obtain a global lower bound $\forall \mu \in \mathcal{D}$, we rely on:

- a patching of local lower bounds $\tilde{\beta}_{\mu^*}(\mu)$ to the surrogate inf-sup stability factors $\beta_{\mu^*}(\mu)$, each one upon a given $\mu^* \in \mathcal{S}$, and
- the evaluation of the stability factor $\beta_h(u_h(\mu^*))$ for (possibly few) selected points $\mu^* \in \mathcal{S}$.

These goals can be achieved through a suitable extension of the SCM algorithm to the case of nonlinear operators, which is detailed in Section 4.1. To estimate the sharpness of the lower bound $\tilde{\beta}_{\mu^*}(\mu) \leq \beta_{\mu^*}(\mu)$ some theoretical results, shown in the following section, are needed.

---

4 From a practical standpoint, we solve the following symmetrized eigenvalue problem,

$$
\frac{1}{2} \left[ \mathbb{F}(\mu^*)^T X^{-1} \mathbb{F}(\mu) + \mathbb{F}(\mu)^T X^{-1} \mathbb{F}(\mu^*) \right] v = \lambda_{\mu^*}(\mu) \mathbb{F}(\mu^*)^T X^{-1} \mathbb{F}(\mu^*) v.
$$

(35)
4.1 Construction of a local lower bound $\tilde{\beta}_{\mu^*}(\mu)$

Let us consider a given $\mu^* \in \mathcal{S}$, and the local lower bound

$$\tilde{\beta}_{\mu^*}(\mu) := \inf_{v \in V_h} \frac{dA(u_h(\mu); \mu)(v, T^* u)}{\|T^* u\|_V^2} = \inf_{v \in V_h} \frac{(T^* u, T^* v)_V}{\|T^* v\|_V^2}. \quad (37)$$

As in the linear case [26], we can show that, for $\mu$ near $\mu^*$, $\tilde{\beta}_{\mu^*}(\mu)$ is a second-order accurate approximation to $\beta_{\mu^*}(\mu)$, according to the following (see Appendix A.2 for the proof)

**Proposition 4** Under the assumptions of Proposition 2, the following relations hold:

$$\tilde{\beta}_{\mu^*}(\mu) - 1 = O(|\mu - \mu^*|) \quad \text{as} \quad \mu \to \mu^*, \quad (38)$$

$$\beta_{\mu^*}(\mu) - \tilde{\beta}_{\mu^*}(\mu) = O(|\mu - \mu^*|^2) \quad \text{as} \quad \mu \to \mu^*. \quad (39)$$

In particular, (38) guarantees that, for $\mu$ near $\mu^*$, the bilinear form $\pi : V \times V \to \mathbb{R}$

$$\pi(u, v) = (T^* u, T^* v)_V$$

is coercive. In other words, the eigenvalues obtained by solving (36) are positive. Thus, we can compute a lower (and an upper) bound to $\tilde{\beta}_{\mu^*}(\mu)$ by applying the SCM algorithm proposed in [14], since this surrogate problem is coercive thanks to (34). The upper bound is evaluated in order to add (successive) constraints, to ensure that the local lower bound is positive (see Step 3 of the procedure below).

Given $\mu^* \in \mathcal{D}$ and a very rich training sample $\Xi_{\text{train}} \subset \mathcal{D}$, we note that $\tilde{\beta}_{\mu^*}(\mu)$ can be seen as the solution of a linear program (recall that $Q_A = Q_a + Q_c$):

$$\tilde{\beta}_{\mu^*}(\mu) = \inf_{y \in \mathcal{Y}_*} J(y; \mu), \quad J(y; \mu) = \sum_{q=1}^{Q_a} \Theta_q^a(\mu)y_q + \sum_{q'=1}^{Q_c} \Theta_q^{c'}(\mu)y_{Q_a+q'}, \quad (40)$$

with $y = (y_1, \ldots, y_{Q_a}, y_{Q_a+1}, \ldots, y_{Q_a+Q_c})$. Here $\mathcal{Y}_* \subset \mathbb{R}^{Q_A}$ is given by $(1 \leq q \leq Q_a, 1 \leq q' \leq Q_c)$:

$$\mathcal{Y}_* = \left\{ y \in \mathbb{R}^{Q_A} : \exists w_h^y \in V_h \left| y_q = \frac{a_q(w_{h_q}^y, T^* w_{h_q}^y)}{\|T^* w_{h_q}^y\|_V^2}, \quad q \leq Q_a \right. \right.,$n

$$\mathcal{Y}_* = \left. \left. \left| y_{Q_a+q'} = \frac{d_{q'}(u_h(\mu^*))(w_{h_q}^y, T^* w_{h_q}^y)}{\|T^* w_{h_q}^y\|_V^2} \right. \right. \right\}.$$

We now build a lower bound through a sequence of suitable relaxed problems of (40) by seeking the minimum of $J$ on a descending sequence of larger sets, built by adding successively linear constraints. We also build an upper bound to $\tilde{\beta}_{\mu^*}(\mu)$, which will serve to define a suitable error indicator in the greedy procedure for the construction of the local lower bound.

To this goal, let us consider the following steps.
1. **Bounding box construction.** In order to guarantee a priori that (40) is well-posed, we can construct a (continuity) bounding box $B_{\mu^*} \subset \mathbb{R}^{Q_A}$ given by [12]

$$B_{\mu^*} = \prod_{q=1}^{Q} \left[ -\frac{\gamma_y^a}{\beta_h(u_h(\mu^*)))}, \frac{\gamma_y^a}{\beta_h(u_h(\mu^*))} \right] \times \prod_{q=1}^{Q} \left[ -\frac{\gamma_y^d(\mu^*)}{\beta_h(u_h(\mu^*)))}, \frac{\gamma_y^d(\mu^*)}{\beta_h(u_h(\mu^*))} \right]$$

(41)

where $\beta(\mu^*)$ is the solution of (27) computed for $\mu = \mu^*$. Alternatively, as recently proposed in [29], we can consider the following bounding box,

$$B_{\mu^*}^Y = \prod_{q=1}^{Q} \left[ \inf_{v \in V_h} \frac{a_q(v, T_{\mu^*} v)}{\|T_{\mu^*} v\|_V^2}, \sup_{v \in V_h} \frac{a_q(v, T_{\mu^*} v)}{\|T_{\mu^*} v\|_V^2} \right] \times \prod_{q=1}^{Q} \left[ \inf_{v \in V_h} \frac{d_q(u_h(\mu^*))(v, T_{\mu^*} v)}{\|T_{\mu^*} v\|_V^2}, \sup_{v \in V_h} \frac{d_q(u_h(\mu^*))(v, T_{\mu^*} v)}{\|T_{\mu^*} v\|_V^2} \right]$$

(42)

which is proved to be tighter than (41), i.e. $B_{\mu^*}^Y \subset B_{\mu^*}$. Let us remark however that the computation of $B_{\mu^*}^Y$ requires additional operations, in particular: (i) for each $\mu^*$ the bounding box has to be fully recomputed, while for the former we can compute the $\gamma_y^a$’s once and for all, and only update the $\gamma_y^d$’s at each iteration; (ii) for each $\mu^*$, $B_{\mu^*}^Y$ requires to compute not only the maximum but also the minimum eigenvalue of the involved bilinear forms. This is a demanding task, which can become unaffordable when $Q_a$ and $Q_c$ become too large. In Section 6 we will show a detailed comparison of these two options.

2. **Relaxed LP problem.** Given a SCM sample $C_{\mu^*} = \{\mu_1^*, \ldots, \mu_k^* \}$ associated to $\mu^*$, compute the corresponding lower bounds $\tilde{\beta}_{\mu^*}(\mu')$, by solving (35) $\forall \mu' \in C_{\mu^*}$; then, define the relaxation set

$$\mathcal{Y}^L_{\mu^*}(C_{\mu^*}) = \{ y \in B_{\mu^*} : J(y; \mu') \geq \tilde{\beta}_{\mu^*}(\mu'), \ \forall \mu' \in C_{\mu^*} \}$$

by selecting a set of additional linear constraints associated to $C_{\mu^*}$. Let us remark that the desired local lower bound $\tilde{\beta}_{\mu^*}^L(\mu)$ is provided by the solution of the following relaxed problem:

$$\tilde{\beta}_{\mu^*}^L(\mu) = \inf_{y \in \mathcal{Y}^L_{\mu^*}(C_{\mu^*})} J(y; \mu), \ \forall \mu \in D_{\mu^*}$$

(43)

since $\tilde{\beta}_{\mu^*}(\mu) \geq \tilde{\beta}_{\mu^*}^L(\mu)$. In fact, $\mathcal{Y}^L_{\mu^*} \subset \mathcal{Y}^L_{\mu^*}(C_{\mu^*})$ and thus the minimum is taken over a larger set. Note that (43) has to be solved $\forall \mu \in \mathcal{X}_{\text{train}}$, whereas the definition of $D_{\mu^*} \subset \mathcal{D}$ will be made precise later on. We can also define an upper bound to $\tilde{\beta}_{\mu^*}(\mu)$ as follows:

$$\tilde{\beta}_{\mu^*}^U(\mu) = \inf_{y \in \mathcal{Y}^U_{\mu^*}(C_{\mu^*})} J(y; \mu), \ \forall \mu \in D_{\mu^*}$$

(44)

where

$$\mathcal{Y}^U_{\mu^*}(C_{\mu^*}) = \{ \bar{y} \in \mathbb{R}^{Q_A} : \bar{y} = \arg \min_{y \in \mathcal{Y}_*} J(y; \mu'), \ \forall \mu' \in C_{\mu^*} \}.$$
we can summarize the whole procedure as follows:

and the actual covered set (global) greedy procedure

by adding the point \( \hat{\mu} \) such that

\[
\hat{\mu} = \arg \max_{\mu \in E_{\mu^*}^{train}} \rho(\mu; C_{\mu^*}), \quad \rho(\mu; C_{\mu^*}) = \frac{\beta_{\mu^*}^{UB}(\mu) - \tilde{\beta}_{\mu^*}^{LB}(\mu)}{\tilde{\beta}_{\mu^*}^{UB}(\mu)},
\]

until the largest ratio is \( \rho(\mu; C_{\mu^*}) \leq \varepsilon_* \), i.e. under a chosen SCM tolerance \( \varepsilon_* \in (0, 1) \). Here we restrict the search for the maximum of \( \rho(\cdot; \cdot) \) to a suitable neighborhood \( E_{\mu^*} \) of \( \mu^* \), which shall represent an (empirical) approximation of the coercivity region (see Proposition 4) of the \( \tilde{\beta}_{\mu^*}(\mu) \); further details about the numerical evaluation of \( E_{\mu^*} \) will be given in Section 6.

Thus, we end up with \( K = |C_{\mu^*}| \) constraints and a local lower bound \( \tilde{\beta}_{\mu^*}^{LB}(\mu) \).

**4.2 Computation of a global lower bound**

In order to turn the local lower bound \( \tilde{\beta}_{\mu^*}^{LB}(\mu) \), computed upon each selected value \( \mu^* \), into a global lower bound, we consider a greedy procedure such as the one addressed in [14,12] for the linear case. For this reason, we refer the reader to those works for further details. We only remark that the output of the coverage procedure are the set \( S = \{\mu^1, \ldots, \mu^J\} \), \( J \leq J_{\max} \) and the associated SCM samples \( C_{\mu^*} \), for any \( j = 1, \ldots, J \), where \( K(j) : = |C_{\mu^j}| < K_{\max} \) is the number of constraints points related to each \( \mu^j \in S \). Thus, a global lower bound for \( \beta(\mu) \) is

\[
\beta^{LB}(\mu) = \beta(\mu^{\sigma(\mu)})^{\tilde{\beta}_{\mu^{\sigma(\mu)}}^{LB}}, \quad \text{being} \quad \sigma(\mu) = \arg \max_{j \in \{1, \ldots, J\}} \beta(\mu^j)^{\tilde{\beta}_{\mu^{\sigma(\mu)}}^{LB}}(\mu),
\]

so that the subdomains \( D_{\mu^j} \) are defined as

\[
D_{\mu^j} = \{\mu \in D : \beta(\mu^{\sigma(\mu)})^{\tilde{\beta}_{\mu^{\sigma(\mu)}}^{LB}}(\mu) \geq \beta(\mu^j)^{\tilde{\beta}_{\mu^j}^{LB}}(\mu), \quad \forall j' = 1, \ldots, J\}.
\]

We also remark that the global lower bound \( \beta^{LB}(\mu) \) interpolates \( \beta(\mu) \) at each \( \mu^* \in S \), being \( \beta^{LB}(\mu^*) = \beta(\mu^*) \). The set \( S = \{\mu^1, \ldots, \mu^J\} \) is built through a (global) greedy procedure, which encapsulates the local ones used for building each SCM sample. By defining the covered set

\[
R_j = \{\mu \in \Xi_{\text{train}} | \beta_{\mu^{\sigma(\mu)}}^{LB}(\mu) > 0 \text{ and } \rho(\mu; C_{\mu^*}) \leq \varepsilon_*\}
\]

and the actual covered set

\[
R_j^{act} = \{\mu \in \Xi_{\text{train}} | \beta_{\mu^{\sigma(\mu)}}^{LB}(\mu; C_{\mu^*}) > 0 \text{ and } \rho(\mu; C_{\mu^*}) \leq \varepsilon_*\},
\]

we can summarize the whole procedure as follows:

---

**Nonlinear SCM algorithm**

**Input:** train sample \( \Xi_{\text{train}}, J_{\max}, K_{\max}, \text{SCM tolerance } \varepsilon_* \), starting point \( \mu^{1*} \).
set $J = 1$, $\mathcal{C}_{\mu^\star} = \{\mu^\star\}$
compute $\beta_h(\mu^\star)$ by (27) and the bounding box $B_{\mu^\star}$

while $J < J_{\text{max}}$, $\Xi_{\text{train}} \neq \emptyset$ and $\rho(\mu) > \epsilon_*$ do
  compute $\beta_{LB} B_{\mu^J}(\mu)$, $\beta_{UB} B_{\mu^J}(\mu)$ and the actual covered set $R_{act}^j$
  compute $\beta_{LB}(\mu)$ as in (45)
  if $R_{act}^j \setminus R_j = \emptyset$ or $|\mathcal{C}_{\mu^J}| = K_{\text{max}}$ do
    update $\Xi_{\text{train}} = \Xi_{\text{train}} \setminus R_j$
    set $J = J + 1$ and select a new $\mu^J$
    compute $\beta_h(\mu^J)$ by (27) and the bounding box $B_{\mu^J}$
  set $\mathcal{C}_{\mu^J} = \{\mu^J\}$ and compute the covered set $R_j$
  else do
    $\hat{\mu} = \arg \max_{\mu \in E_{\mu^J}} \rho(\mu; \mathcal{C}_{\mu^J})$
    set $\mathcal{C}_{\mu^J} = \mathcal{C}_{\mu^J} \cup \{\hat{\mu}\}$
    compute $\beta_{LB}^J(\hat{\mu})$ by solving (35)
    set $R_j = R_{act}^j$
  end if
end while

Let us highlight which are the main computational costs of this problem. We denote by $n_{\text{train}} = |\Xi_{\text{train}}|$ and we define
$$
n_{\beta} = \sum_{j=1}^{J} |\mathcal{C}_{\mu^J}|, \quad n_C = \max_{j=1,\ldots,J} |\mathcal{C}_{\mu^J}|.
$$

In the offline stage we have to: (i) solve $n_{\beta}$ times problem (19) in order to compute $u_h(\mu)$ and assemble the Fréchet derivative; (ii) solve $n_{\text{eig}}^1 = n_{\beta} + Q_a + JQ_c$ (respectively $n_{\text{eig}}^2 = n_{\beta} + 2JQ_a + 2JQ_c$) eigenproblems when using the bounding box (41) (respectively (42)), (iii) solve $n_{\text{train}} n_{\beta}$ linear programs to compute the current global lower bounds (45) at each iteration of the algorithm.

In the online stage, each evaluation $\mu \rightarrow \beta_h(u_h(\mu))$ only requires to solve $J$ linear programs in $Q_a$ variables with at most $n_C + 2Q_A$ constraints (independently of the employed bounding box).

Remark 2 As in the linear case, the computational complexity of the offline stage of the SCM depends inherently on $Q_A N_h$, where the dependence on the dimension $N_h$ is due to eigenvalues calculation. Thus, already for rather small problems, the size $Q_A$ of the affine expansion may cause the Offline stage to become potentially very expensive. A two-level affine decomposition strategy was recently proposed in [16] to tackle the case of large affine operators (e.g. recovered through the empirical interpolation method).

5 Heuristic strategies

Even if the online stage can be performed very rapidly, the rather expensive offline stage, required to construct local lower bounds to $\beta_h(u_h(\mu))$, might compromise the efficiency of the whole SCM procedure. Furthermore, from our direct experience, we remark a rather slow convergence when dealing with many ($p \geq 3$)
parameters (see also Section 6). For these reasons, seeking alternative strategies to compute a lower bound of stability factors becomes mandatory in the case of nonlinear operators, depending on many parameters. In this section we show some possible heuristic strategies devised to meet an efficiency requisite (at both the offline and the online stages), which return reliable surrogates: (i) a first strategy based on the numerical minimization of the stability factor over the parameter range; (ii) a second one based on a suitable radial basis interpolation procedure. Both these strategies require the evaluation of the high fidelity stability factor $\beta_h(\mu)$; hence, let us recall that, fixed $\mu \in D$, the computation of $\beta_h(\mu)$ requires to solve the following eigenvalue problem: find $(\lambda(\mu), v) \in \mathbb{R}_+ \times V_h, v \neq 0$ such that

$$F(\mu)^T X^{-1} F(\mu) v = \lambda(\mu) X v,$$

(47)

where the matrix $X$ is defined by (25), whereas the stability factor is $\beta_h(\mu) = \sqrt{\lambda_{\text{min}}(\mu)}$.

5.1 Minimum stability factor

In this case, we approximate the (\mu-dependent) stability factor $\beta_h(\mu)$ by a constant, given by the global minimum of the stability factor itself, i.e.

$$\beta_{\text{LB}} = \min_{\mu \in D} \beta_h(\mu).$$

(48)

Since $\beta_h(\mu)$ is possibly a non-convex function of $\mu$, in order to find its global minimum on $D$ we employ a local active-set optimization solver combined with a multi-start globalization strategy, see e.g. [10] for further details. Unless provided by the user, optimization solvers usually employ finite differences approximations to reconstruct the gradient of the function to be minimized, thus requiring several function evaluations. Since in our case the evaluation of the function $\beta_h(\mu)$ is rather expensive, we seek for an explicit expression of its sensitivities with respect to the parameters, i.e. the partial derivatives with respect to the parameters components.

Let us denote with $A(\mu) = F(\mu)^T X^{-1} F(\mu)$ the left-hand side matrix in (47). The eigenvalue derivatives with respect to the parameters components can be easily computed by differentiating (47) and then premultiplying both sides by the corresponding eigenvector:

$$\frac{\partial \lambda(\mu)}{\partial \mu_j} = \frac{v(\mu)^T A_{\mu_j}(\mu) v(\mu)}{\|v(\mu)\|^2_X},$$

being $A_{\mu_j}(\mu) = \partial A(\mu)/\partial \mu_j$. Since $\beta_h(\mu) = \sqrt{\lambda_{\text{min}}(\mu)}$, we obtain

$$\frac{\partial \beta_h(\mu)}{\partial \mu_j} = \frac{1}{2\beta_h(\mu)} \frac{v(\mu)^T (F_{\mu_j}(\mu)^T X^{-1} F(\mu) + F(\mu)^T X^{-1} F_{\mu_j}(\mu)) v(\mu)}{\|v(\mu)\|^2_X}.$$

Thus, the sensitivity of the stability factor with respect to $\mu_j$ can be calculated from the stability factor itself, the corresponding eigenvector $\lambda_j$ and the sensitivity of the problem matrix with respect to $\lambda_j$. If the problem is linear, exploiting the affinity assumptions (11), we could easily compute the matrix sensitivities as
Rigorous and heuristic strategies for the approximation of stability factors

\[ F_{\mu_j}(\mu) = \sum_{q=1}^{Q} \frac{\partial \psi_q(\mu)}{\partial \mu_j} \psi_q. \]

On the other hand, when the problem features a quadratic nonlinearity, the problem matrix depends on the solution itself, i.e. \( F(\mu) = F(u_h(\mu); \mu) \). Therefore,

\[ F_{\mu_j}(u_h(\mu); \mu) = F\left( \frac{\partial u_h(\mu)}{\partial \mu_j}; \mu \right) + F_{\mu_j}(u; \mu)|_{u_h(\mu)}. \]

In order to avoid the expensive computation of the solution sensitivities \( \frac{\partial u_h(\mu)}{\partial \mu_j} \), we approximate \( F_{\mu_j}(u_h(\mu); \mu) \approx F_{\mu_j}(u; \mu)|_{u_h(\mu)} \). In other words, we assume that, in a suitable neighborhood of \( \mu \), the matrix sensitivities are more influenced by the explicit parameters dependence than by the implicit one through \( u_h(\mu) \).

This approach shows to be effective when the stability factor has moderate variations with respect to parameters, for instance in the case of geometrical ones. However, as soon as the dimension of the parameter space increases, the computation of a global minimum becomes extremely expensive, since the multi-start strategy requires to perform the optimization for a large number of initial guesses. Moreover, this procedure ensures to find a rigorous lower bound, although it might be, in principle, not very tight if \( \beta_h(\mu) \) experiments strong variations on the parameter set \( D \). For these reasons, an alternative strategy based on a suitable interpolation procedure [18, 20] for the approximation of the whole function \( \beta_h(\mu) \) seems more promising.

5.2 Interpolant of the stability factor

Let us denote by \( \Xi_{\text{fine}} \subset D \) a sample set whose dimension \( n_{\text{fine}} = |\Xi_{\text{fine}}| \) is sufficiently large. We (arbitrarily and a priori) select a (possibly small) set of interpolation points \( \Xi_I \subset \Xi_{\text{fine}} \) and compute the stability factor \( \beta_h(\mu) \), by solving the eigenproblem (47), for each \( \mu \in \Xi_I \). Then, we compute a suitable interpolant surrogate \( \beta_I(\mu) \) such that

\[ \beta_I(\mu) = \beta_h(\mu) \quad \forall \mu \in \Xi_I \quad \text{and} \quad \beta_I(\mu) > 0 \quad \forall \mu \in \Xi_{\text{fine}}. \]

Depending on the number of parameters and their range of variation, different interpolation methods might be employed. In [20] we used a simple linear interpolant and an equally spaced grid of interpolation points since the parameter space was only two dimensional. When the parameter space show higher dimensions the use of uniform grids immediately suffers the curse of dimensionality, requiring to compute the true stability factor in a huge number of interpolation points. For this reason, following [18], we propose to use an interpolatory radial basis function (RBF) technique, which is known to be particularly suited for interpolation from scattered data in high-dimension spaces (for a general introduction to RBF methods see, e.g., [3]).

In order to avoid negative values of the interpolant \( \beta_I(\mu) \), we first perform the interpolation on a starting grid \( \Xi_I \) of interpolation points and we evaluate the resulting interpolant on the fine grid. Then, we enrich the interpolation grid.
Ξ_I by adding further interpolation points in the regions where the interpolant is negative, thus yielding the greedy procedure reported below:

**Positive RBF-surrogate**

**Input:** the evaluation grid Ξ_fine, a set of n_0^I samples Ξ_I, n_{max}

(i) **Build initial coarse interpolation**
   for j = 1 : n_0^I
       set µ_j = Ξ_I(j) and assemble the matrix F(µ_j)
       compute β_h(µ_j) by solving the eigenvalue problem (47)
   end for

(ii) **Ensure the positivity of the interpolant**
   evaluate β_I(µ) on Ξ_fine and set µ_{neg} = {µ : β_I(µ) ≤ 0}
   while µ_{neg} ≠ ∅ and j < n_{max} do
       set j = j + 1, µ_j = arg min{β_I(µ_{neg})}
       compute β_h(µ_j) by solving the eigenvalue problem (47)
       update the set Ξ_I = Ξ_I ∪ {µ_j}
       build the RBF interpolant β_I(µ)
       evaluate β_I(µ) on Ξ_fine and set µ_{neg} = {µ : β_I(µ) < 0}
   end while

At the end of this procedure, we obtain a surrogate β_I(µ) which is positive (provided that n_{max} is sufficiently large), interpolates exactly β_h(µ) in each µ ∈ Ξ_I, and whose online evaluation for a given µ requires only O(n_I) operations, thus being independent of N_h.

Let us denote by n_{neg} the number of points selected by the second step of the algorithm. In the offline stage, we must (i) solve at most n_0^I + n_{neg} eigenvalue problems; (ii) build 1 + n_{neg} times the RBF interpolant, which requires O(n_I n_fine) operations (being n_I the dimension of the adaptively enriched set Ξ_I); (iii) evaluate n_0^I + n_{neg} times the RBF interpolant, requiring O(n_I n_fine) operations. Note that the first step of the algorithm can be performed in parallel.

Although not always rigorous, this surrogate yields a good approximation of the stability factor, with a much less computational effort with respect to the SCM algorithm (and requiring a less involved, and thus error-prone, implementation). However, as the dimension (and extent) of the parameter space increases, the effectiveness of the procedure highly depends on the initial full factorial grid Ξ_I which is adopted. Indeed, if Ξ_I is too coarse, most of the time is spent in the second step of the algorithm trying to ensure the positivity of the interpolant, and eventually resulting in a poor approximation. On the other hand, if Ξ_I is more refined, the additional computational effort can be often unnecessary, since too many points are added in regions with a slowly varying response. We overcome these inconveniences by further improving this method in the following section.

---

5 Note that the parameters domain D is normalized to the unit hypercube [0, 1]^p for interpolation; in this way, the results of the interpolation are not affected by possible different scales and range of variations of the parameters.
5.3 Interpolant surrogate with adaptive sampling

In order to achieve a compromise between (i) adding new points in locations with a highly varying response, (ii) adding points in unsampled regions of the domain and (iii) ensuring the positivity of the interpolant, we propose an adaptive strategy based on a four-component criterion $C(\mu)$:

$$C(\mu) = (\|\nabla \beta_I(\mu)\| + \epsilon)(\|\Delta \beta_I(\mu)\| + \epsilon) \left( \frac{h(\mu)}{\max h(\mu)} \right)^2 g(\beta_I(\mu))$$  \hspace{1cm} (49)

similarly to what proposed in [17]. Let us describe the role of each factor:

- the first two terms account for local changes of the interpolant\(^6\); an offset parameter $\epsilon$ ensures a non-zero value when $\|\nabla \beta_I(\mu)\| = 0$ or $|\Delta \beta_I(\mu)| = 0$;
- the third and the fourth terms promote the selection of space-filling points and penalize negative values of the interpolant, respectively, being

$$h(\mu) = \min_{\mu_j \in \Xi_I} \|\mu - \mu_j\|_2, \quad g(s) = \begin{cases} 1 & s > 0 \\ \alpha e^{-s} & s \leq 0 \end{cases}$$

and $\alpha > 0$ a tuning parameter to be prescribed.

In this adaptive algorithm, the new sample locations are then selected as the ones which maximize the criterion $C$. The maximum value of $C(\mu)$ is found by evaluating the criterion on the fine grid and extracting the maximum, rather than using a global optimization algorithm. Indeed, evaluating $C$ is fast (complexity of $O(n_{I\text{fine}}^2)$) and can be easily performed in parallel. The adaptive loop is terminated either when a predetermined number of interpolation points have been added or when a required accuracy is reached, and reads as follows:

```
Adaptive RBF-surrogate
Input: the evaluation grid $\Xi_{\text{fine}}$, a set of $n_I^0$ starting samples $\Xi_I$, $n_{\text{max}}$

(i) Build initial coarse interpolation
   for $j = 1 : n_I^0$
      set $\mu_j \in \Xi_I(j)$ and assemble the matrix $F(\mu_j)$
      compute $\beta_h(\mu_j)$ by solving the eigenvalue problem (47)
   end for
   build the RBF interpolant $\beta_I(\mu)$

(ii) Enrich interpolation with adaptive sampling
   evaluate $\beta_I(\mu)$ on $\Xi_{\text{fine}}$
   while $j < n_{\text{max}}$ and $E_j > \text{tol}$ do
      compute criterion $C(\mu)$ as defined in (49)
      set $\mu_j = \arg \max C(\mu)$ and assemble the matrix $F(\mu_j)$
      compute $\beta_h(\mu_j)$ by solving the eigenvalue problem (47)
      build the RBF interpolant $\beta_{I\cup\mu_j}(\mu)$
      evaluate $E_j = \|\beta_I(\mu) - \beta_{I\cup\mu_j}(\mu)\|_\infty$
      update the set $\Xi_I = \Xi_I \cup \{\mu_j\}$
   end while
```

\(^6\) Note that the derivatives of the interpolant are available analytically and, therefore, both the gradient and the Laplacian can be calculated exactly, rather than reconstructed through a numerical approximation.
Several techniques can be employed to quantify this latter and possibly estimate the interpolation error, see [17] and references therein for further details. Here we simply require the $L_\infty(D)$-norm of two consecutive iterates to be under a certain tolerance. Let us remark that the offline costs are just slightly increased with respect to the Positive RBF-surrogate strategy, since the number of operations required to evaluate the criterion $C(\mu)$ only depends on $n_f$ and $n_{\text{fine}}$, being independent of $N_h$.

6 Numerical results and comparisons

In this section we illustrate through some numerical examples the properties and performances of the proposed nonlinear-SCM algorithm and of the heuristic strategies. As a test case we consider a fluid flow over a backward facing step channel [1], described by the steady Navier-Stokes equations:

\begin{align*}
-\nu \Delta u + (u \cdot \nabla)u + \nabla p &= 0 \quad \text{in } \Omega_o(\mu) \\
\text{div } u &= 0 \quad \text{in } \Omega_o(\mu) \\
\Gamma_d^o &= u = \mathbf{g} \quad \text{on } \Gamma_d^o(\mu) \\
\Gamma_w^o &= u = 0 \quad \text{on } \Gamma_w^o(\mu) \\
-p n + \nu (\nabla u) n &= 0 \quad \text{on } \Gamma_n^o(\mu),
\end{align*}

(50)

where $(v, p)$ are the velocity and pressure defined over a parametrized domain $\Omega_o(\mu)$ (see Fig. 1). We denote by $\Gamma_d^o = \Gamma_d^o \cup \Gamma_w^o(\mu)$ the Dirichlet portion of $\partial \Omega_o$, while $\Gamma_w^o(\mu)$ denotes the outflow boundary. We define the Reynolds number as $Re = Du_b/\nu$, where $\nu$ is the kinematic viscosity, $D = 2h$ being $h = 1$ the height of the channel at the inflow, while $u_b = 2/3 \max g = 1$, being $g = 6y(1 - y)$ the inflow profile. We consider $p = 3$ parameters: the Reynolds number $\mu_1 = Re$ (so that $\nu = 2/\mu_1$), the step height $\mu_2$ and the channel length $\mu_3$ (downstream of the step).

Fig. 1: Sketch of the channel geometry with boundaries and partition in affine subdomains. The first subdomain is independent of the parameters, while $\Omega_2 = \Omega_2(\mu_3)$ and $\Omega_3 = \Omega_3(\mu_2, \mu_3)$. Coloring is given by the velocity field magnitude obtained for $Re = 250$.

The weak formulation of problem (50) can be cast under the form (2)-(11) by introducing a decomposition of $\Omega_o(\mu)$ into three subdomains (see Figure 1) and a suitable affine geometrical transformation, so that the problem is mapped to a fixed, reference domain $\Omega$; further details can be found, e.g., in [8,18,28].
Once the problem is formulated as in (2), we introduce its FE discretization. We use a (inf-sup stable) $P_1^b-P_1$ approximation for the velocity and pressure variables, i.e. continuous linear FE enriched by bubble functions for the velocity and continuous linear FE for the pressure, see e.g. [22]. The total number of degrees of freedom is $N_h = 40,064$, obtained using a mesh of 11,485 triangular elements. For the solution of the Navier-Stokes equations, we employ a few Picard iterations followed by some Newton iterations to reach a relative tolerance of $10^{-8}$ on the norm of the increment. Thus, convergence is always achieved for all the range of Reynolds number we consider. To facilitate the computation of extreme eigenvalues, we consider a weighted norm on $V$: for any $v = (v, q) \in V$, $\|v\|_{2, V} := \tilde{a}(v, v; \hat{\mu}) + \lambda \|v\|_{L_2}^2 + \lambda \|q\|_{L_2}^2$, where $\hat{\mu}$ is a reference (prescribed) parameter value, $\tilde{a}(\cdot, \cdot; \mu)$ corresponds to the diffusion term in the momentum equation, while

$$\lambda = \inf_{w \in [H^1(\Omega)]^3} \frac{\tilde{a}(v, w; \hat{\mu})}{\|w\|_{L_2}^2} > 0.$$  

The research code we use in this work has been developed in the Matlab environment; all the linear system solves are performed using the sparse direct solver provided by Matlab, as well as all the eigenvalue problems are solved using Matlab eigs solver. To compute the global minimum, we use the Matlab routines fmincon and MultiStart as local solver and globalization algorithm, respectively. We also take advantage of the existing SCM algorithm already developed (for linear problems) in the rbMIT library [13]. For the RBF-interpolant we always use thin plate splines as radial basis functions. Parallelism is exploited to speed up the matrix assembly in the Navier-Stokes solver as well as to speed up some embarrassingly parallel portions of the algorithms we propose. The reported computational times will mainly serve to compare the different strategies.

6.1 Backward-facing step channel with a physical parameter

As a first test case we consider as parameter only the Reynolds number $\mu_1 \in [20, 250]$, fixing the geometrical parameters $\mu_2 = 1$ and $\mu_3 = 10$. The affine decomposition (11) is recovered for $Q_a = 3$, $Q_c = 1$ and $Q_f = 3$.

First, we numerically verify the inequality proved in Proposition 3,

$$|\beta_h(u_h(\mu)) - \beta_h(u_N(\mu))| \leq 2\gamma_h(\mu)\|u_h(\mu) - u_N(\mu)\|_V. \quad (51)$$

We build the RB space following the procedure described in [19]; requiring a tolerance of $10^{-3}$ on the error estimate (31), we end up with a reduced space of dimension $N = 11$. In Figure 2 we report the graphs of the left- and right-hand sides of (51) with respect to $N$ (computed on a test sample of 20 parameter values and then averaged).

Then, we numerically verify the coercivity property of the local lower bound $\tilde{\beta}_{\mu^*}(\mu)$, i.e. as shown in Theorem 4,

$$\tilde{\beta}_{\mu^*} = 1 + O(|\mu - \mu^*|), \quad \text{as } \mu \rightarrow \mu^*.$$  

For the sake of verification, we select “by hand” $J = 9$ parameter points $\mu^*_j$, $1 \leq j \leq J$, and compute the corresponding local lower bound $\tilde{\beta}_{\mu^*_j}(\mu)$, which are reported in Figure 3. As expected, for each $\mu^*_j$, the local lower bounds $\tilde{\beta}_{\mu^*_j}(\mu)$
Let us now apply the nonlinear-SCM algorithm with a tolerance $\epsilon_\star = 0.7$, $n_{\text{train}} = 1000$ and using the original bounding box (41). The greedy procedure selects $J = 6$ anchor points $\mu_\star$ and $|C_{\mu_\star}|_{j=1}^6 = [3, 3, 6, 2, 2, 1]$ constraints, thus requiring to solve $n_{\text{eig}}^1 = 26$ eigenproblems. In Figure 4 we report the resulting lower bound $\beta_{\text{LB}}(\mu)$ as well as the subdomains partition $D_{\mu_\star}$ induced by the algorithm.

Then, in the same setting, we apply the SCM using the tighter bounding box (41): we obtain $J = 4$, $|C_{\mu_\star}|_{j=1}^4 = [3, 3, 2, 1]$ and $n_{\text{eig}}^2 = 41$. As regards the computational performances, the two options require roughly the same time (about 20 minutes) to be performed. The (tighter) bounding box (42) lead to a sharper lower bound (see Figure 4) yet selecting a smaller number of anchor points $\mu_\star$. However it globally requires to solve a higher number of eigenvalue problems. In particular, its computational complexity is highly dependent on the number of terms $Q_a$ and $Q_c$ in the affine decomposition.
Table 1: Test 1. Comparison of the computational cost of the SCM algorithm when using the bounding box (41) and (42).

<table>
<thead>
<tr>
<th></th>
<th>Bounding box (41)</th>
<th>Bounding box (42)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J$ (number of selected $\mu^*$)</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>$</td>
<td>C_{\mu^*}</td>
<td>, j = 1, \ldots, J$</td>
</tr>
<tr>
<td>Number of eigenproblems</td>
<td>26</td>
<td>41</td>
</tr>
<tr>
<td>SCM tolerance $\epsilon_*$</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>Total time (s)</td>
<td>1088</td>
<td>1191</td>
</tr>
</tbody>
</table>

Fig. 4: Test 1. Comparison between the lower bound of the stability factor obtained using the SCM algorithm with different bounding box: $\beta_{SCM1}$ refers to (41), while $\beta_{SCM2}$ is obtained using (42). We also report the subdomains (46) induced by the algorithm, with different colors (top: $\beta_{SCM1}$, bottom: $\beta_{SCM2}$). The corresponding $\mu^*_j, 1 \leq j \leq J$ are represented by black crosses.

Let us now move to the heuristic strategies. We first compute the minimum stability factor by following the procedure described in Section 5.1: performing the multi-start optimization with three different initial points, we find a minimum stability factor $\beta_{LB} = 0.1025$ (attained for $\mu_1 = 250$), which turns out to be the global minimum of $\beta_h(\mu)$ (see Figure 5). In this case, the algorithm requires to solve 45 eigenproblems. We highlight the importance of the multi-start strategy; indeed, if we start the optimization from $\mu_1 < 40$, the algorithm converges to the local minimum attained for $\mu_1 = 20$, thus largely overestimating the global one.

Then, we compute the adaptive RBF-surrogate $\beta_1(\mu)$: starting from an initial coarse grid of 4 (uniformly distributed) interpolation points, the adaptive procedure selects 10 additional interpolation points to achieve a model accuracy of $10^{-3}$. The adaptivity criterion demonstrates to be effective, since most of the interpolation points are added in the region with the highest variation of $\beta_h(\mu)$, as it can be seen in Figure 5. In this case the construction of the surrogate only requires to solve 14 eigenproblems, taking about 8 minutes.
Let us remark that, while the final interpolant is almost coincident with the exact stability factor, already the initial one (computed on the coarse grid) can be considered as a satisfactory approximation for our purposes.

![Graph](image)

**Fig. 5:** Test 1. Comparison of the heuristic strategies. The value of the minimum stability factor and the interpolant surrogate with respect to the true stability factor $\beta_h(\mu)$ (black line) are reported. For the latter, both the initial interpolation on a coarse grid of 4 points (blue dashed line) and the result of the adaptive strategy (red line) are shown.

<table>
<thead>
<tr>
<th># eigenproblems</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nonlinear-SCM with (41)</td>
<td>26</td>
</tr>
<tr>
<td>Nonlinear-SCM with (42)</td>
<td>41</td>
</tr>
<tr>
<td>Minimum</td>
<td>45</td>
</tr>
<tr>
<td>Adaptive RBF-surrogate</td>
<td>14</td>
</tr>
</tbody>
</table>

**Table 2:** Test 1. Comparison of the computational costs. Computations have been performed using 4 cores on a desktop computer.

### 6.2 Backward-facing step channel with both physical and geometrical parameters

As a second test we consider as parameters both the Reynolds number $\mu_1$ and the height of the channel step $\mu_2$: the parameter space is now given by $\mathcal{D} = [20, 200] \times [0.5, 1.5]$. We have slightly restricted the range of the parameter $\mu_1$ to avoid numerical instabilities occurring for high values of $\mu_1$ and $\mu_2$. The affine decomposition (11) now holds with $Q_a = 5$ and $Q_c = 2$.

We first run the nonlinear-SCM algorithm with a tolerance $\epsilon_* = 0.85$, $n_{\text{train}} = 10^4$ and using the original bounding box (41). The algorithm shows an extremely low convergence: $J = 195$ parameter values $\mu^{1*}, \ldots, \mu^{J*}$ and about 750 constraints are selected, requiring to solve $O(10^3)$ eigenproblems. It turns out that the poor convergence rate of the SCM is mainly due to the geometrical variation induced by the parameter $\mu_2$. Indeed, as a confirmation, we have run the SCM freezing the first parameter $\mu_1 = 100$: the computation of the lower bound (shown in Figure 7) required $J = 17$ parameters value $\mu^{1*}$, a rather large number compared to the results of the previous section.
6.3 Backward-facing step channel with three parameters

We now let all the three parameters free to change; in particular the parameter domain is now given by \( \mathcal{D} = [20, 200] \times [0.5, 1.5] \times [9, 12] \), while \( Q_a = 9 \) and \( Q_c = 4 \). From the results of the previous section, it is rather clear that both the nonlinear-SCM and the minimum stability factor strategies are no more viable options in this case. The adaptive RBF-surrogate represents the only chance to obtain, with a reasonable (and somehow predictable) computational effort, a satisfactory approximation of the stability factor.
Fig. 8: Test 2. Comparison between the interpolant surrogate $\beta_I(\mu)$ (top) and the true stability factor $\beta_h(\mu)$ (bottom). For the former, we report also the initial coarse grid (magenta) and the interpolation points (black) selected by the adaptive procedure.

We start the interpolation on a coarse grid of $3 \times 3 \times 3$ uniformly distributed interpolation points, and then we let the adaptive procedure select additional 18 samples. We report in Figure 9 the resulting approximation of the stability factor; once again, the adaptive criterion promotes the selection of interpolation points in the regions with highest variations of $\beta_h(\mu)$. In Figure 10, we compare the true stability factor and the surrogate $\beta_I(\mu)$ in the setting of the first test case, i.e. as functions of $\mu_1$, with $\mu_2 = 1$ and $\mu_3 = 10$ fixed; in the same figure we also report the interpolant surrogate obtained using further 20 interpolation points (adaptively selected).

Once again, the adaptive procedure correctly places the interpolation points in the most varying regions, so that a tight approximation can be easily obtained with a moderate computational effort. In particular, while the SCM algorithm tends to place the control points $\mu^*$ in regions where the solution – rather than the stability factor – is more sensitive to changes in the parameters, the adaptive interpolation is only affected by the parametric response of the stability factor, so that the overall computational efficiency of the latter is much higher.

7 Further remarks and conclusions

In this paper we have developed some numerical strategies for the rapid and reliable evaluation of lower bounds of stability factors related to parametrized, inf-sup stable, nonlinear operators. We have taken advantage of a well-known theoretical framework to show some results allowing to extend the Successive Constraint Method to the case of nonlinear operators. This methodology features an Offline/Online decomposition and is well-suited for stability analysis and a posteriori error estimation in a RB context. In particular, the extension presented in this paper features a relevant novelty, being completely uncoupled from the generation of the reduced approximation space, thanks to a further theoretical result, of which we also provide a numerical assessment.
Even if the online stage can be performed very rapidly, the rather expensive offline stage of the SCM procedure might compromise the efficiency of the whole reduction process. Furthermore, the convergence of the SCM algorithm is rather slow already in the linear case, when dealing with $p \geq 3$ parameters, and this is even more dramatic in the nonlinear case, as shown by the numerical results presented in this work. For these reasons, we have developed some alternative, heuristic strategies to compute a lower bound of the stability factor. Two possible algorithms devised to meet an efficiency requisite (at both the offline and the online stages), yet returning reliable surrogates, have been introduced in this work. They are based on either piecewise constant approximations of the stability factor (obtained through numerical optimization procedures), or the construction of
suitable radial basis interpolants (through an adaptive choice of the interpolation points in the parameter space).

The former provides rigorous lower bounds and it is rather effective when the stability factor has moderate variations with respect to the parameters, and the dimension of the parameter space is very small. The latter returns very tight approximations to the stability factor, with a small computational effort. Moreover, the interpolation grid can be adaptively enriched, through a suitable greedy procedure, by adding further interpolation points in the regions where the interpolant is negative, or where the stability factor experiences stronger variations. Several numerical results show the effectivity of this procedure, which in our opinion represents a break-through for the sake of efficiency of error estimation related to reduced-order approximations. In this way, the very time-consuming Offline calculation required by SCM can be greatly reduced, without getting rid of the effectivity of the lower bounds. Furthermore, the interpolation-based heuristic procedure can be easily performed in parallel, thus yielding a further computational reduction of the Offline stage, as well as of the whole construction of a RB approximation, even when dealing with other (than quadratical) nonlinear operators.

A Appendix

A.1 Proof of Proposition 2

From the definition (5) of $dA(\cdot; \mu)(\cdot, \cdot)$ and the affine decomposition (11), we have that

$$dA(u(\mu); \mu)(v, w) - dA(u(\mu^*); \mu^*)(v, w) = \sum_{q=1}^{Q_u} (\Theta_u^q(\mu) - \Theta_u^q(\mu^*)) a_q(v, w) + \sum_{q=1}^{Q_u} \Theta_u^q(\mu) d_q(u(\mu), v, w) - \sum_{q=1}^{Q_u} \Theta_u^q(\mu^*) d_q(u(\mu^*), v, w).$$

The first term can be easily bounded as

$$|I| \leq Q_u L_u |\mu - \mu^*| |\gamma_u||w||V|,$$

where $L_u = \max_{q=1, \ldots, Q_u} L_u^q$, being the $L_u^q$'s the Lipschitz constants of the functions $\Theta_u^q(\cdot)$, while $\gamma_u = \max_{q=1, \ldots, Q_u} \gamma_u^q$, being the $\gamma_u^q$'s the continuity constants of the bilinear forms $a_q(\cdot, \cdot)$. Let us now rewrite the second term as

$$|II| \leq Q_u L_u \|\mu - \mu^*\|_\mathbb{R} \|\gamma_d||v||V| \|w||V| + Q_u M_d^u \|\gamma_d||v||w||V| \|u(\mu) - u(\mu^*)\|_V.$$

Here $L_u = \max_{q=1, \ldots, Q_u} L_u^q$, being the $L_u^q$'s the Lipschitz constants of the functions $\Theta_u^q(\cdot)$, $\gamma_d$ is the larger among the continuity constants of the trilinear forms $d_q(\cdot, \cdot, \cdot)$, and

$$M_d^u = \max_{\mu \in \mathbb{D}} \max_{q=1, \ldots, Q_u} \Theta_u^q(\mu).$$

Since the solution $u(\mu)$ of problem (2) is bounded for every $\mu \in \mathbb{D}$ – thanks to (15) – and Lipschitz continuous with respect to $\mu$ (see Theorem 1), there exist positive constants $K_u$ and $L_u$ such that

$$\|u(\mu)\|_V \leq K_u, \quad \|u(\mu) - u(\mu^*)\|_V \leq L_u |\mu - \mu^*|,$$
Furthermore, we have
\[ |\Pi| \leq \left( L_c K_u + M^e_D L_u \right) Q_c \gamma_d |\mu - \mu^*| \|v\|_V \|w\|_V. \] 
(53)

Combining (52) and (53), in the end we obtain (16) with constant
\[ C = Q_c L_a \gamma_a + Q_c \gamma_d \left( L_c K_u + M^e_D L_u \right). \]

Furthermore, we have
\[
\|T^\mu w - T^\mu^* w\|^2_V = (T^\mu w - T^\mu^* w, T^\mu w - T^\mu^* w)_V \\
= dA(u(\mu); \mu)(v, T^\mu w - T^\mu^* w) - dA(u(\mu^*); \mu^*)(v, T^\mu w - T^\mu^* w) \\
\leq C |\mu - \mu^*| \|w\|_V \|T^\mu w - T^\mu^* w\|_V \\
\leq \frac{C}{\beta(\mu^*)} |\mu - \mu^*| \|T^\mu w - T^\mu^* w\|_V \\
\]
by exploiting (16) and (10), from which we obtain, for \( \mu \neq \mu^* \),
\[
\|T^\mu w - T^\mu^* w\|_V \leq \frac{C}{\beta(\mu^*)} |\mu - \mu^*| \|T^\mu w\|_V \quad \forall w \in V.
\]

A.2 Proof of Proposition 4

In order to show (38), we start by observing that
\[
(T^\mu v, T^\mu^* v)_V = (T^\mu v - T^\mu^* v + T^\mu^* v, T^\mu^* v)_V = \|T^\mu^* v\|^2_V + (T^\mu v - T^\mu^* v, T^\mu^* v)_V \\
= \|T^\mu^* v\|^2_V + dA(u_h(\mu); \mu)(v, T^\mu^* v) - dA(u_h(\mu^*); \mu^*)(v, T^\mu^* v)
\]
so that
\[
\beta_{\mu^*}(\mu) \leq 1 + \inf_{v \in V_h} \frac{|dA(u_h(\mu); \mu)(v, T^\mu^* v) - dA(u_h(\mu^*); \mu^*)(v, T^\mu^* v)|}{\|T^\mu^* v\|^2_V}.
\]

In order to bound this quantity, we exploit the result (16) of Proposition 2, which is valid for any \( v_h, w_h \in V_h \), too (see e.g. [4, Remark 13.2]). Thus, for any \( v \in V_h \),
\[
|dA(v(\mu); \mu)(v, T^\mu^* v) - dA(v(\mu^*); \mu^*)(v, T^\mu^* v)| \\
\leq C |\mu - \mu^*| \|v\|_V \|T^\mu^* v\|_V \leq \frac{C}{\beta(\mu^*)} |\mu - \mu^*| \|T^\mu^* v\|^2_V,
\]
so that
\[
\beta_{\mu^*}(\mu) \leq 1 + \frac{C}{\beta(\mu^*)} |\mu - \mu^*| \]
or, equivalently, \( \beta_{\mu^*}(\mu) - 1 = O(|\mu - \mu^*|) \) as \( \mu \to \mu^* \). In order to show (39), we first expand
\[
\beta_{\mu^*}^2(\mu) = \inf_{v \in V_h} \|T^\mu v\|^2_V = \inf_{v \in V_h} \frac{(T^\mu v + (T^\mu v - T^\mu^* v), T^\mu v + (T^\mu v - T^\mu^* v))_V}{\|T^\mu v\|^2_V} \\
= 1 + \inf_{v \in V_h} \frac{2 (T^\mu v - T^\mu^* v, T^\mu^* v)_V}{\|T^\mu v\|^2_V} + \|T^\mu w - T^\mu^* w\|^2_V.
\]
Thanks to (17), we have
\[
\frac{\|T^\mu w - T^\mu^* w\|^2_V}{\|T^\mu^* w\|^2_V} \leq \frac{C^2}{\beta^2(\mu^*)} |\mu - \mu^*|^2,
\]
and by recognizing that
\[
\inf_{v \in V_h} \frac{(T^\mu v - T^\mu^* v, T^\mu^* v)_V}{\|T^\mu^* v\|^2_V} = \beta_{\mu^*}(\mu) - 1,
\]
we end up with \( \beta_{\mu^*} (\mu) = -1 + 2\beta_{\mu^*} (\mu) + O(|\mu - \mu^*|^2) \) as \( \mu \to \mu^* \). By taking the square root, using a Taylor series expansion and the fact that \( O(\beta_{\mu^*} (\mu) - 1) = O(|\mu - \mu^*|) \) thanks to (38), we obtain:

\[
\sqrt{-1 + 2\beta_{\mu^*} (\mu) + O(|\mu - \mu^*|^2)} = \\
= 1 + \frac{1}{2} (2\beta_{\mu^*} (\mu) - 2 + O(|\mu - \mu^*|^2)) - \frac{1}{8} (2\beta_{\mu^*} (\mu) - 2 + O(|\mu - \mu^*|^2))^2 + O(|\mu - \mu^*|^3) \\
= 1 + (\beta_{\mu^*} (\mu) - 1) - \frac{1}{2} (\beta_{\mu^*} (\mu) - 1)^2 + O(|\mu - \mu^*|^3),
\]

so that

\[
\beta_{\mu^*} (\mu) = \beta_{\mu^*} (\mu) + O(\beta_{\mu^*} (\mu) - 1)^2 + O(|\mu - \mu^*|^3), \quad \text{as} \quad \mu \to \mu^*.
\]

Finally, by exploiting again (38), we end up with (39). □

Acknowledgements

We thank Dr. Masayuki Yano (MIT) for several useful discussions and Prof. Anthony T. Patera (MIT) for his initial encouragement in deepening the analysis of the methodologies presented in this paper. We are also grateful to Prof. Alfio Quarteroni (EPFL) for his valuable comments and suggestions. This work has been supported in part by the Swiss National Science Foundation (Project 141034) and by the SHARM 2012-2014 SISSA post-doctoral research grant on the project “Reduced Basis Methods for shape optimization in computational fluid dynamics”.

References

Recent publications:

MATHEMATICS INSTITUTE OF COMPUTATIONAL SCIENCE AND ENGINEERING
Section of Mathematics
Ecole Polytechnique Fédérale
CH-1015 Lausanne

40.2013 D. Kressner, M. Steinlechner, A. Uschmajew:
Low-rank tensor methods with subspace correction for symmetric eigenvalue problems

41.2013 A. Abdulle, O. Budác:
An adaptive finite element heterogeneous multiscale method for Stokes flow in porous media

42.2013 C. Jäggli, L. Iapichino, G. Rozza:
An improvement on geometrical parametrizations by transfinite maps

43.2013 P. Pacciarini, G. Rozza:
Stabilized reduced basis method for parametrized advection-diffusion PDEs

44.2013 A. Koshakji, A. Quarteroni, G. Rozza:
Free form deformation techniques applied to 3D shape optimization problems

45.2013 J. E. Catrillon-Candas, F. Nobile, R. F. Tempone:
Analytic regularity and collocation approximation for PDEs with random domain deformations

01.2014 Giovanni Migliorati:
Multivariate Markov-type and Nikolskii-type inequalities for polynomials associated with downward closed multi-index sets

02.2014 Federico Negri, Andrea Manzoni, Gianluigi Rozza:
Certified reduced basis method for parametrized optimal control problems governed by the Stokes equations

03.2014 Cedric Effenberger, Daniel Kressner:
On the residual inverse iteration for nonlinear eigenvalue problems admitting a Rayleigh functional

04.2014 Takahito Kashiwabara, Claudia M.Colciago, Luca Dedè, Alfio Quarteroni:
Numerical Well-posedness, regularity, and convergence analysis of the finite element approximation of a generalized Robin boundary value problem

05.2014 Björn Adlerborn, Bo Kagström, Daniel Kressner:
A parallel QZ algorithm for distributed memory HPC systems

06.2014 Michele Benzi, Simone Deparis, Gwenol Grandperrin, Alfio Quarteroni:
Parameter estimates for the relaxed dimensional factorization preconditioner and application to hemodynamics

07.2014 Assyr Abdulle, Yun Bai:
Reduced order modelling numerical homogenization

08.2014 Andrea Manzoni, Federico Negri:
Rigorous and heuristic strategies for the approximation of stability factors in nonlinear parametrized PDEs