5. DIFFERENT APPROACH TO COMPUTE THE COHERCIVITY FUNCTION $\alpha(\mu)$

Recall the error estimator $\eta(\mu) := \dfrac{\|\hat{v}_\delta(\mu)\|_V}{\alpha_\delta(\mu)}$

→ To compute η we need $\alpha_\delta(\mu)$ for each value of the parameters

Computed cheaply with online / offline phase

This part, cohercivity function, DIFFICULT TO COMPUTE

→ The smallest eigenvalue of the spectral problem

Associated to high fidelity problem $\forall \mu \in P$ $\alpha(\mu) = \inf_{v_\delta \in V_\delta} \dfrac{a(v_\delta, v_\delta; \mu)}{\|v_\delta\|_V^2}$

So this is an eigenvalue problem

$\Rightarrow \lambda_{min} = \alpha(\mu)$ cohercivity $\quad a(w_\delta, v_\delta; \mu) = \lambda (w_\delta, v_\delta)_V \quad \forall v_\delta \in V_\delta$ smallest eigenvalue

$\lambda_{max} = \gamma(\mu)$ continuity

How we can compute the eigenvalue? We fix the basis $\{\varphi_i\}$ in V_δ and we have this linear combination $w_\delta = \sum_{j=1}^{N_\delta} w_{\delta,j}\varphi_j$, so that we obtain the matrix form

$\boxed{A_\delta^\mu = \lambda X_\delta w_\delta}$, where $(X_\delta)_{ij} = \langle \varphi_i, \varphi_j \rangle_V$ is the scalar product matrix, and $w_\delta = \begin{pmatrix} w_{\delta,1} \\ w_{\delta,N_\delta} \end{pmatrix}$

↳ is the mass inner product matrix in V

↳ Generalized eigenvalue problem

But this is very costly, so this has to be modified

Different approach to compute $\alpha(\mu)$ - Most of this approach approximate $\alpha(\mu)$ from the bottom computing $\alpha_{lower\text{-}bound}(\mu) \le \alpha(\mu)$

We want now to analize these different approaches.

5.1 Min-ϑ approach

→ This method works for the affine case and for the so called **parametrically cohercive problems**

We need * The affinity hypothesis (separate the parameter part to the other)

Hypothesis $\boxed{a(u, v; \mu) = \sum_{q=1}^{Q_a} \vartheta_a^q(\mu) a_q(u, v)}$

* The positivity hypothesis

$\boxed{\vartheta_a^q(\mu) > 0 \quad \forall \mu \in P}$

* A sort of cohercivity

$\boxed{a_q(u, v) \ge 0 \quad \forall \delta \in V_\delta}$

So that here we can compute, selecting one parameter $\tilde{\mu} \in P$, $\alpha(\tilde{\mu})$ ⇒ let's assume we know $\alpha(\tilde{\mu})$ for one value of $\tilde{\mu} \in P$

By definition, it holds $\alpha(\mu) := \inf_{v_\delta \in V_\delta} \dfrac{a(v_\delta, v_\delta; \mu)}{\|v_\delta\|_V^2}$

$$= \inf_{v_\delta \in V_\delta} \sum_{q=1}^{Q_a} \vartheta_a^q(\mu) \frac{a_q(v_\delta, v_\delta)}{\|v_\delta\|_V^2} \cdot \frac{\vartheta_a^q(\tilde{\mu})}{\vartheta_a^q(\tilde{\mu})} \ge \inf_{v_\delta \in V_\delta} \min_{1 \le q \le Q_a} \frac{\vartheta_a^q(\mu)}{\vartheta_a^q(\tilde{\mu})} \boxed{\sum_{q=1}^{Q_a} \vartheta_a^q(\tilde{\mu}) \frac{a_q(v_\delta, v_\delta)}{\|v_\delta\|_V^2}}$$

$= \dfrac{a(v_\delta, v_\delta, \tilde{\mu})}{\|v_\delta\|_V^2}$

$$= \min_{1 \le q \le Q_a} \frac{\vartheta_a^q(\mu)}{\vartheta_a^q(\tilde{\mu})} \boxed{\inf_{v_\delta \in V_\delta} \frac{a(v_\delta, v_\delta; \tilde{\mu})}{\|v_\delta\|_V^2}} = \alpha(\tilde{\mu}) \text{ computed in offline phase}$$

Hence, we obtain $\boxed{\alpha(\mu) \ge \min_{1 \le q \le Q_a} \dfrac{\vartheta_a^q(\mu)}{\vartheta_a^q(\tilde{\mu})} \alpha(\tilde{\mu}) := \alpha_{LB}(\mu)}$ ⇒ lower bound of α

↓ Extending this idea we can build

5.2 Multi-parameter min-ϑ approach

→ This case requires to know $I \ge 1$ values of $\alpha(\tilde{\mu}_i)$, $i \in \{1, .., I\}$, and following the previous analize we can obtain

Select $I \ge 1 \Rightarrow \{\tilde{\mu}_i\}_{i=1,..,I}$ so that we can compute $\{\alpha(\tilde{\mu}_i)\}_{i=1,..,I}$

Hence,

$$\boxed{\alpha(\mu) \ge \max_{1 \le i \le I} \left(\min_{1 \le q \le Q_a} \frac{\vartheta_a^q(\mu)}{\vartheta_a^q(\tilde{\mu}_i)} \alpha(\tilde{\mu}_i) \right) := \alpha_{LB}(\mu)}$$

5.3 Successive constraint method SCM

: complex, but works very properly → based on the definition of a functional

Idea: build a functional $S: P \times Y \to \mathbb{R}$ where $Y \subseteq \mathbb{R}^{Q_a}$, $y \in Y := [y_1, .., y_{Q_a}]$ → affine element $\in \mathbb{R}$

$$S(\mu, y) := \sum_{q=1}^{Q_a} \vartheta_a^q(\mu) y_q$$

To better see the relation between the bilinear form and the functional, we can define the space:

$$\boxed{\mathcal{Y} := \left\{ y \in \mathbb{R}^{Q_a} : \exists v_\delta \in V_\delta \ \text{s.t.}\ y_q = \frac{a_q(v_\delta, v_\delta)}{\|v_\delta\|_V^2},\ q = \{1, \dots, Q_a\} \right\}}$$

Hence, the coercivity can be seen as

$$\boxed{\alpha(\mu) := \inf_{v_\delta \in V_\delta} \frac{a(v_\delta, v_\delta; \mu)}{\|v_\delta\|_V^2} = \min_{y \in \mathcal{Y}} S(\mu; y)} \quad \forall \mu \in P$$

equivalent problem to avoid to compute the eigenvalue problem

But, instead of solving the minimization problem in $\mathcal{Y}$, we solve it in $\mathcal{Y}_{LB}$ so that the problem is easier to solve it

In a bigger set $\mathcal{Y}_{LB}$, I can achieve a better solution

How can we find $\mathcal{Y}_{LB}$? The method suggest to find $\underset{\alpha_{LB} \le \alpha \le \alpha_{UB}}{\mathcal{Y}_{UB} \subseteq \mathcal{Y} \subseteq \mathcal{Y}_{LB}}$, so that we can compute

$0 \le \alpha_{LB}(\mu) \le \alpha_{UB}(\mu) \Rightarrow \boxed{0 \le 1 - \frac{\alpha_{LB}(\mu)}{\alpha_{UB}(\mu)} < 1}$

NB In a smaller set $\mathcal{Y}_{UB}$, I have a bigger constant α_{UB} because α is the solution of the minimization problem, so that in $\mathcal{Y}_{UB} \subseteq \mathcal{Y}$ could not exist a smaller value of α_{UB}

Measure of the quality of approximation of $\alpha(\mu)$

Moreover, we define $\eta_\alpha(\mu) := 1 - \frac{\alpha_{LB}(\mu)}{\alpha_{UP}(\mu)}$

Define a sequence of sets $\dots \subseteq \mathcal{Y}_{UB}^{n-1} \subseteq \mathcal{Y}_{UB}^{n} \subseteq \mathcal{Y} \subseteq \mathcal{Y}_{LB}^{n} \subseteq \mathcal{Y}_{LB}^{n-1} \subseteq \dots \ \forall n \ge 1$

Idea: iterative algorithm to improve the value with goal $\mathcal{Y}_{UB} \subseteq \mathcal{Y} \subseteq \mathcal{Y}_{LB}$

Algorithm -
1. Selecting $P_a \subseteq P$, take $\mu_1 \in P_a$ and exactly compute $\alpha(\mu_1)$ with eigenvalue problem
2. Compute w_δ's eigenfunction associated to $\alpha(\mu_1)$

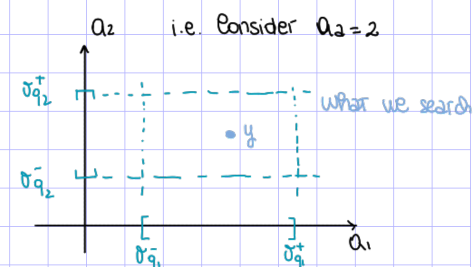
   $a(w_\delta, v_\delta) = \lambda (w_\delta, v_\delta)_V \quad \forall v_\delta \in V_\delta$

   $$\boxed{\mathcal{Y}_{UP}^1 = \left\{ y^1 \in \mathbb{R}^{Q_a} : y_q^1 = \frac{a_q(w_\delta^1, v_\delta^1)}{\|v_\delta\|},\ 1 \le q \le Q_a \right\}}$$ this is an approximation of $\mathcal{Y}$, an upper bound approxim.

3. $\forall q \in \{1, \dots, Q_a\}$ we have to solve $a_q(w_\delta, v_\delta) = \lambda (w_\delta, v_\delta)_V \quad \forall v_\delta \in V_\delta$ obtaining the eigenvalue
   - $\sigma_q^- :=$ smallest eigenvalue
   - $\sigma_q^+ :=$ largest eigenvalue

   $$\boxed{\mathcal{Y}_{LB}^1 := \prod_{q=1}^{Q_a} [\sigma_q^-, \sigma_q^+]}$$

   i.e. Consider $Q_a = 2$



   Hence, here we build $\mathcal{Y}_{UB}^1 \subseteq \mathcal{Y} \subseteq \mathcal{Y}_{LB}^1$ because we select $\mu \in P$

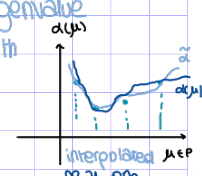
4. We have to improve with the recursion step
   - we have $\{\mu_1, \dots, \mu_n\} \in P_a \subset P$
   - we have $\underset{\alpha_{UB}(\mu)}{\mathcal{Y}_{UB}^n} \subseteq \mathcal{Y} \subseteq \underset{\alpha_{LB}(\mu)}{\mathcal{Y}_{LB}^n}$
   - we compute $\eta_\alpha(\mu) := 1 - \frac{\alpha_{LB}(\mu)}{\alpha_{UB}(\mu)}$ → Is this lower than a tolerance? $\eta < \varepsilon$
   - If so, we have to enlarge the spaces: select $\mu_{n+1} \in P$ and compute $\{\alpha(\mu_{n+1}), w_\delta^{n+1}\}$

     $$\boxed{\mathcal{Y}_{UB}^{n+1} = \left\{ y^{n+1} \in \mathbb{R}^{Q_a} : y_q^{n+1} = \frac{a_q(w_\delta^{n+1}, w_\delta^{n+1})}{\|w_\delta\|_V^2} \quad \forall q \in \{1, \dots, Q_a\} \right\}}$$

   - So that $\mathcal{Y}_{UB}^{n+1} := \mathcal{Y}_{UB}^n \cup \mathcal{Y}^{n+1}$
   - $\forall q \in \{1, \dots, Q_a\}$ solve the eigenvalue problem $a_q(w_\delta, v_\delta) = \lambda (w_\delta, v_\delta) \quad \forall v_\delta \in V_\delta$
   - Compute $B = \prod_{q=1}^{Q_a} [\sigma_q^-, \sigma_q^+]$ → solving the eigenvalue problem computing min e max eigenvalue

     $$\boxed{\mathcal{Y}_{LB}^{n+1} := \left\{ y \in B : \sum_{q=1}^{Q_a} \sigma_q^a(\mu)\, y \ge \alpha(\mu) \quad \forall \mu \in \{\mu_1, \dots, \mu_{n+1}\} \right\}}$$

     The minimization is a linear programming minimization problem with at most $2Q_a + (n+1)$ constraints for Q_a variable



5.4 Interpolation approach

→ The idea is to interpolate the function $\alpha(\mu)$ with $\underset{=}{\alpha}$ over a subset $P_I \subseteq P$

Consider $\underset{=}{\alpha} : P \to \mathbb{R}$, $\mu \mapsto \alpha(\mu)$, we can select $P_I \subset P$ and $\{\mu_i\} \subset P_I$ we can compute the real value of $\alpha(\mu_i)$

I as small as possible

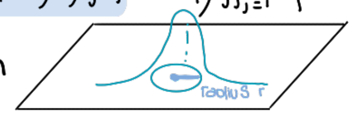
After computing all the exact value, we interpolate the point to compute our $\alpha(\mu)$

But interpolation in huge space is not so good ← not polynomial one

⇒ CURSE OF DIMENSIONALITY: to avoid we use RBF (exponential function)

↳ we cannot use the "classic" polynomial interpolation, as it suffers of the curse of dimensionality, i.e. the number of interpolant points is too large when P increase
⇒ To avoid this we use radial basis function

In particular, we seek for the logarithm of $\alpha_I(\mu)$, s.t. $\log(\alpha_I(\mu)) = w_0 + w^T\mu + \sum_{i=0}^{N_I} \gamma_i \phi_i(||\mu-\mu_j||)$ in $\{\mu_j\}_{j=1}^{N_I}$ points belong to P s.t. $\log(\alpha_I(\mu_j)) = \log(\alpha(\mu_j))$ and s.t. $\sum_{i=0}^{N_I}\gamma_i = 0$, $\sum_{i=0}^{N_I}\gamma_i\mu_i^p = 0 \;\forall p\in[1,\dots,P]$ where $\phi: P\to\mathbb{R}$, $r\mapsto e^{r^2}$ is the RBF, i.e. the gaussian



Algorithm 1 Successive Constraint Method (SCM)

Idea. Build a functional $S(\mu, y)$ where $y \in Y : S(\mu, y) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) y_p$

*Define the **space***

$$Y := \{y \in \mathbb{R}^{Q_a} : \exists v_\delta \in V_\delta \; s.t. \; y_q = \frac{a_q(v_\delta, v_\delta)}{||v_\delta||_V^2}, \; q = \{1, \dots, Q_a\}\}$$

*The **coercivity** is*

$$\alpha(\mu) \geq \inf_{v_\delta \in V_\delta} \frac{a(v_\delta, v_\delta; \mu)}{||v_\delta||_V^2}, \forall \mu \in P$$

Equivalent problem. *Instead of solving in Y, we solve it in a cheaper way in Y_{UB} and Y_{LB}*

Goal. *Find $\alpha_{LB} \in Y_{LB}$ and $\alpha_{UB} \in Y_{UB}$ to bound the solution $\alpha \in Y$. using an **iterative algorithm***

Quality measure. $\eta_\alpha = 1 - \frac{\alpha_{LB}(\mu)}{\alpha_{UB}(\mu)}$

Input.

- Set of candidate functions P_a
- Tolerance ϵ

Initialize. Take $\mu_1 \in P_a$ and exactly compute $\alpha(\mu_1)$ by solving the eigenvalue problem

Compute w_δ^1 eigenfunction associated to $\alpha(\mu_1)$

Define

$$Y_{UB}^1 = \{y^1 \in \mathbb{R}^{Q_a} : y_q^1 = \frac{a_q(w_\delta^1, v_\delta^1)}{||v_\delta||}, 1 \leq q \leq Q_a\}$$

$\forall q \in \{1, \dots, Q_a\}$ solve the eigenvalue problem $\forall v_\delta \in V_\delta$

$$a_q(w_\delta, v_\delta) = \lambda(w_\delta, v_\delta)_V$$

obtaining $\theta_q^- :=$ the smallest eigenvalue and $\theta_q^+ :=$ the biggest eigenvalue

Build $Y_{LB}^1 = \prod_{q=1}^{Q_a} [\theta_q^-, \theta_q^+]$

while $\eta_\alpha(\mu) > \epsilon$ **do**

Compute $\eta_\alpha(\mu) = 1 - \frac{\alpha_{LB}(\mu)}{\alpha_{UB}(\mu)}$

Enlarge *the space.* Select $\mu_{n+1} \in P_a$ and compute $\alpha(\mu_{n+1}; w_\delta^{n+1})$

Define

$$Y_{UB}^{n+1} = \{y^{n+1} \in \mathbb{R}^{Q_a} : y_q^{n+1} = \frac{a_Q(w_\delta^{n+1}, w_\delta^{n+1})}{||w_\delta||_V^2} \; \forall q \in 1, \dots, Q_a\}$$

$\rightarrow Y_{UB}^{n+1} = Y_{UB}^n \cup Y^{n+1}$

$\forall q \in \{1, \dots, Q_a\}$ solve the eigenvalue problem $\forall v_\delta \in V_\delta$

$$a_q(w_\delta, v_\delta) = \lambda(w_\delta, v_\delta)_V$$

obtaining $\theta_q^- :=$ the smallest eigenvalue and $\theta_q^+ :=$ the biggest eigenvalue

Build $B = \prod_{q=1}^{Q_a} [\theta_q^-, \theta_q^+]$

$$Y_{LB}^{n+1} = \{y^{n+1} \in B : \sum_{q=1}^{Q_a} \Theta_q^a(\mu) y \geq \alpha(\mu) \; \forall \mu \in \{\mu_1, \dots, \mu_{n+1}\}\}$$

end while

Output. Approximate solution α