Quelques exemples d'algorithmes gloutons :

leur analyse et leurs applications pour la simulation numérique de phénomènes de la mécanique.

Yvon Maday[1]

[1]Laboratoire Jacques-Louis Lions
Université Pierre et Marie Curie, Paris, France
and Brown Univ.

Rennes, ENS campus Ker Lann

Outline

Outline

Interpolation is a general and classical tool for approximation

Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^M \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Background.

General issues about interpolation

Interpolation is a general and classical tool for approximation
Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^M \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Background.

General issues about interpolation

Interpolation is a general and classical tool for approximation
Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^M \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Background.

General issues about interpolation

Interpolation is a general and classical tool for approximation
Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^{M} \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Background.

General issues about interpolation

Interpolation is a general and classical tool for approximation
Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^M \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Interpolation is a general and classical tool for approximation
Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^M \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Interpolation is a general and classical tool for approximation
Assume you are given a set $\varphi_1, \ldots, \varphi_n, \ldots$ of linearly independent functions, given a function f that you want to approximate, the problem is

- find a familly of scalars $\{\alpha_n^M\}_{1 \le n \le M}$ such that
- $f(\zeta_m^M) = \sum_{n=1}^{M} \alpha_n^M \varphi_n(\zeta_m^M)$

where the *interpolation points* ζ_m^M are suitably chosen
of course, the project is generally not limited to the approximation in the only

$$X_M = \text{Span}\{\varphi_n,\ 1 \le n \le M\}$$

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Background.

General issues about interpolation

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Among the classical questions raised by interpolation

- given a set of points, does the interpolant at these points exist;
- is the interpolant unique;
- how does the interpolation process compares with other approximations (in particular orthogonal projections);
- is there an optimal selection for the interpolation points;
- is there a constructive optimal selection for the interpolation points;

These question are covered in the polynomial case, though not completely and the answers are complex and rather recent

Outline

Definition

Let $\mathcal{X}$ be a normed linear space, X be a subset of $\mathcal{X}$ and X_n be a generic n-dimensional subspace of $\mathcal{X}$. The deviation of X from X_n is

$$E(X; X_n) = \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}}.$$

The *Kolmogorov n-width* of X in X is given by

$$\begin{aligned} d_n(X, \mathcal{X}) &= \inf\{E(X; X_n) : X_n \text{ an } n\text{-dimensional subspace of X}\} \\ &= \inf_{X_n} \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}} \ . \end{aligned} \tag{1}$$

The n-width of X thus measures the extent to which X may be approximated by a n-dimensional subspace of $\mathcal{X}$.

Definition

Let $\mathcal{X}$ be a normed linear space, X be a subset of $\mathcal{X}$ and X_n be a generic n-dimensional subspace of $\mathcal{X}$. The deviation of X from X_n is

$$E(X; X_n) = \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}}.$$

The *Kolmogorov n-width* of X in X is given by

$$\begin{aligned} d_n(X, \mathcal{X}) &= \inf\{E(X; X_n) : X_n \text{ an } n\text{-dimensional subspace of X}\} \\ &= \inf_{X_n} \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}} \ . \end{aligned} \tag{1}$$

The n-width of X thus measures the extent to which X may be approximated by a n-dimensional subspace of $\mathcal{X}$.

Approximation in a space of small n-width.

Interpolation

We are looking for a constructive way of approximating in $\mathcal{X}$, we assume that $\mathcal{X} \subset \mathcal{C}^0$

We propose a greedy approach both for constructing the interpolation points and the discrete spaces X_M,

our method is hierarchical

Approximation in a space of small n-width.

Interpolation

We are looking for a constructive way of approximating in $\mathcal{X}$, we assume that $\mathcal{X} \subset \mathcal{C}^0$

We propose a greedy approach both for constructing the interpolation points and the discrete spaces X_M,

our method is hierarchical

Approximation in a space of small n-width.

Interpolation

We are looking for a constructive way of approximating in $\mathcal{X}$, we assume that $\mathcal{X} \subset \mathcal{C}^0$

We propose a greedy approach both for constructing the interpolation points and the discrete spaces X_M,

our method is hierarchical

Outline

Approximation in a space of small n-width.

definition of the magic points

The first interpolating function is

$$\varphi_1 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi\|_{L^\infty(\Omega)}$$

The first interpolation point is

$$\zeta_1 = \arg \max_{x \in \Omega} |\varphi_1|$$

and we set $q_1 = \varphi_1(\cdot)/\varphi_1(\zeta_1)$ The second interpolating function is

$$\varphi_2 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi(\cdot) - \Phi(\zeta_1)q_1\|_{L^\infty(\Omega)}$$

The second interpolation point is

$$\zeta_2 = \arg \max_{x \in \Omega} |\varphi_2(\cdot) - \varphi_2(\zeta_1)q_1|$$

and we set $q_2 = \varphi_2(\cdot) - \varphi_2(\zeta_1)q_1/\varphi_2(\zeta_2) - \varphi_2(\zeta_1)q_1(\zeta_2)$ and we proceed by induction

Approximation in a space of small n-width.

definition of the magic points

The first interpolating function is

$$\varphi_1 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi\|_{L^\infty(\Omega)}$$

The first interpolation point is

$$\zeta_1 = \arg \max_{x \in \Omega} |\varphi_1|$$

and we set $q_1 = \varphi_1(\cdot)/\varphi_1(\zeta_1)$ The second interpolating function is

$$\varphi_2 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi(\cdot) - \Phi(\zeta_1)q_1\|_{L^\infty(\Omega)}$$

The second interpolation point is

$$\zeta_2 = \arg \max_{x \in \Omega} |\varphi_2(\cdot) - \varphi_2(\zeta_1)q_1|$$

and we set $q_2 = \varphi_2(\cdot) - \varphi_2(\zeta_1)q_1/\varphi_2(\zeta_2) - \varphi_2(\zeta_1)q_1(\zeta_2)$ and we proceed by induction

Approximation in a space of small n-width.

definition of the magic points

The first interpolating function is

$$\varphi_1 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi\|_{L^\infty(\Omega)}$$

The first interpolation point is

$$\zeta_1 = \arg \max_{x \in \Omega} |\varphi_1|$$

and we set $q_1 = \varphi_1(\cdot)/\varphi_1(\zeta_1)$ The second interpolating function is

$$\varphi_2 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi(\cdot) - \Phi(\zeta_1)q_1\|_{L^\infty(\Omega)}$$

The second interpolation point is

$$\zeta_2 = \arg \max_{x \in \Omega} |\varphi_2(\cdot) - \varphi_2(\zeta_1)q_1|$$

and we set $q_2 = \varphi_2(\cdot) - \varphi_2(\zeta_1)q_1/\varphi_2(\zeta_2) - \varphi_2(\zeta_1)q_1(\zeta_2)$ and we proceed by induction

Approximation in a space of small n-width.

definition of the magic points

The first interpolating function is

$$\varphi_1 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi\|_{L^\infty(\Omega)}$$

The first interpolation point is

$$\zeta_1 = \arg \max_{x \in \Omega} |\varphi_1|$$

and we set $q_1 = \varphi_1(\cdot)/\varphi_1(\zeta_1)$ The second interpolating function is

$$\varphi_2 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi(\cdot) - \Phi(\zeta_1)q_1\|_{L^\infty(\Omega)}$$

The second interpolation point is

$$\zeta_2 = \arg \max_{x \in \Omega} |\varphi_2(\cdot) - \varphi_2(\zeta_1)q_1|$$

and we set $q_2 = \varphi_2(\cdot) - \varphi_2(\zeta_1)q_1/\varphi_2(\zeta_2) - \varphi_2(\zeta_1)q_1(\zeta_2)$ and we proceed by induction

Approximation in a space of small n-width.

definition of the magic points

The first interpolating function is

$$\varphi_1 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi\|_{L^\infty(\Omega)}$$

The first interpolation point is

$$\zeta_1 = \arg \max_{x \in \Omega} |\varphi_1|$$

and we set $q_1 = \varphi_1(\cdot)/\varphi_1(\zeta_1)$ The second interpolating function is

$$\varphi_2 = \arg \max_{\Phi \in \mathcal{S}_1[X]} \|\Phi(\cdot) - \Phi(\zeta_1)q_1\|_{L^\infty(\Omega)}$$

The second interpolation point is

$$\zeta_2 = \arg \max_{x \in \Omega} |\varphi_2(\cdot) - \varphi_2(\zeta_1)q_1|$$

and we set $q_2 = \varphi_2(\cdot) - \varphi_2(\zeta_1)q_1/\varphi_2(\zeta_2) - \varphi_2(\zeta_1)q_1(\zeta_2)$ and we proceed by induction

Approximation in a space of small n-width.

The recursion formula

We thus construct, by induction, the nested sets of basis functions $\{q_1, \ldots, q_M\}$ and the nested sets of interpolation points $T_M = \{\zeta_1, \ldots, \zeta_M\}, 1 \leq M \leq M_{\max},$.

For $M = 3, \ldots, M_{\max}$, we first solve the interpolation problem for $\alpha_j^{M-1}(\Phi), 1 \leq j \leq M-1$, from

$$\sum_{j=1}^{M-1} q_j(\zeta_i)\alpha_j^{M-1}(\Phi) = \Phi(\zeta_i), \quad i = 1, \ldots, M-1 \,, \tag{2}$$

and compute

$$\mathcal{I}_{M-1}[\Phi] = \sum_{j=1}^{M-1} \alpha_j^{M-1}(\Phi) q_j \,, \tag{3}$$

and

$$\varepsilon_{M-1}(\Phi) = \|\Phi(\cdot) - \mathcal{I}_{M-1}[\Phi(\cdot)]\|_{L^\infty(\Omega)} \,, \tag{4}$$

for all $\Phi \in X$;

Approximation in a space of small n-width.

The recursion formula

We thus construct, by induction, the nested sets of basis functions $\{q_1, \ldots, q_M\}$ and the nested sets of interpolation points $T_M = \{\zeta_1, \ldots, \zeta_M\}, 1 \leq M \leq M_{\max},$.
For $M = 3, \ldots, M_{\max}$, we first solve the interpolation problem for $\alpha_j^{M-1}(\Phi), 1 \leq j \leq M-1$, from

$$\sum_{j=1}^{M-1} q_j(\zeta_i)\alpha_j^{M-1}(\Phi) = \Phi(\zeta_i), \quad i = 1, \ldots, M-1\,, \tag{2}$$

and compute

$$\mathcal{I}_{M-1}[\Phi] = \sum_{j=1}^{M-1} \alpha_j^{M-1}(\Phi) q_j\,, \tag{3}$$

and

$$\varepsilon_{M-1}(\Phi) = \|\Phi(\cdot) - \mathcal{I}_{M-1}[\Phi(\cdot)]\|_{L^\infty(\Omega)}\,, \tag{4}$$

for all $\Phi \in X$;

Approximation in a space of small n-width.

The recursion formula

We thus construct, by induction, the nested sets of basis functions $\{q_1, \ldots, q_M\}$ and the nested sets of interpolation points $T_M = \{\zeta_1, \ldots, \zeta_M\}, 1 \leq M \leq M_{\max},$.
For $M = 3, \ldots, M_{\max}$, we first solve the interpolation problem for $\alpha_j^{M-1}(\Phi), 1 \leq j \leq M-1$, from

$$\sum_{j=1}^{M-1} q_j(\zeta_i)\alpha_j^{M-1}(\Phi) = \Phi(\zeta_i), \quad i = 1, \ldots, M-1 \ , \tag{2}$$

and compute

$$\mathcal{I}_{M-1}[\Phi] = \sum_{j=1}^{M-1} \alpha_j^{M-1}(\Phi) q_j \ , \tag{3}$$

and

$$\varepsilon_{M-1}(\Phi) = \|\Phi(\cdot) - \mathcal{I}_{M-1}[\Phi(\cdot)]\|_{L^\infty(\Omega)} \ , \tag{4}$$

for all $\Phi \in X$;

Approximation in a space of small n-width.

The recursion formula

We thus construct, by induction, the nested sets of basis functions $\{q_1, \ldots, q_M\}$ and the nested sets of interpolation points $T_M = \{\zeta_1, \ldots, \zeta_M\}, 1 \leq M \leq M_{\max},$.
For $M = 3, \ldots, M_{\max}$, we first solve the interpolation problem for $\alpha_j^{M-1}(\Phi), 1 \leq j \leq M-1$, from

$$\sum_{j=1}^{M-1} q_j(\zeta_i)\alpha_j^{M-1}(\Phi) = \Phi(\zeta_i), \quad i = 1, \ldots, M-1 \ , \tag{2}$$

and compute

$$\mathcal{I}_{M-1}[\Phi] = \sum_{j=1}^{M-1} \alpha_j^{M-1}(\Phi) q_j \ , \tag{3}$$

and

$$\varepsilon_{M-1}(\Phi) = \|\Phi(\cdot) - \mathcal{I}_{M-1}[\Phi(\cdot)]\|_{L^\infty(\Omega)} \ , \tag{4}$$

for all $\Phi \in X$;

Approximation in a space of small n-width.

The recursion formula

We thus construct, by induction, the nested sets of basis functions $\{q_1, \ldots, q_M\}$ and the nested sets of interpolation points $T_M = \{\zeta_1, \ldots, \zeta_M\}, 1 \leq M \leq M_{\max},$.
For $M = 3, \ldots, M_{\max}$, we first solve the interpolation problem for $\alpha_j^{M-1}(\Phi), 1 \leq j \leq M-1$, from

$$\sum_{j=1}^{M-1} q_j(\zeta_i)\alpha_j^{M-1}(\Phi) = \Phi(\zeta_i), \quad i = 1, \ldots, M-1 \ , \tag{2}$$

and compute

$$\mathcal{I}_{M-1}[\Phi] = \sum_{j=1}^{M-1} \alpha_j^{M-1}(\Phi) q_j \ , \tag{3}$$

and

$$\varepsilon_{M-1}(\Phi) = \|\Phi(\cdot) - \mathcal{I}_{M-1}[\Phi(\cdot)]\|_{L^\infty(\Omega)} \ , \tag{4}$$

for all $\Phi \in X$;

We then define

$$\varphi_M = \arg \max_{\Phi \in \mathcal{S}_1[X]} \varepsilon_{M-1}(\Phi) \, , \tag{5}$$

and

$$\zeta_M = \arg \max_{x \in \Omega} \|\varphi_M(x) - \mathcal{I}_{M-1}[\varphi_M(x)])\|_{L^\infty(\Omega)} \, , \tag{6}$$

we finally set $q_M(x) = \frac{\varphi_M - \mathcal{I}_{M-1}[\varphi_M]}{\varphi_M(\zeta_M) - \mathcal{I}_{M-1}[\varphi_M](\zeta_M)}$ and $B^M_{ij} = q_j(x_i), 1 \leq i, j \leq M$.

The procedure is well posed if X is of sufficently large dimension (for $M \leq M_{max} \leq \dim X$).

Note that the matrix B^M is invertible and lower triangular (the diagonal is Id).

Approximation in a space of small n-width.

The recursion formula

We then define

$$\varphi_M = \arg \max_{\Phi \in \mathcal{S}_1[X]} \varepsilon_{M-1}(\Phi) \, , \tag{5}$$

and

$$\zeta_M = \arg \max_{x \in \Omega} \|\varphi_M(x) - \mathcal{I}_{M-1}[\varphi_M(x)])\|_{L^\infty(\Omega)} \, , \tag{6}$$

we finally set $q_M(x) = \frac{\varphi_M - \mathcal{I}_{M-1}[\varphi_M]}{\varphi_M(\zeta_M) - \mathcal{I}_{M-1}[\varphi_M](\zeta_M)}$ and $B^M_{ij} = q_j(x_i), 1 \leq i, j \leq M$.

The procedure is well posed if X is of sufficently large dimension (for $M \leq M_{max} \leq \dim X$).

Note that the matrix B^M is invertible and lower triangular (the diagonal is Id).

Approximation in a space of small n-width.

The recursion formula

We then define

$$\varphi_M = \arg \max_{\Phi \in \mathcal{S}_1[X]} \varepsilon_{M-1}(\Phi) \, , \tag{5}$$

and

$$\zeta_M = \arg \max_{x \in \Omega} \|\varphi_M(x) - \mathcal{I}_{M-1}[\varphi_M(x)])\|_{L^\infty(\Omega)} \, , \tag{6}$$

we finally set $q_M(x) = \frac{\varphi_M - \mathcal{I}_{M-1}[\varphi_M]}{\varphi_M(\zeta_M) - \mathcal{I}_{M-1}[\varphi_M](\zeta_M)}$ and $B^M_{ij} = q_j(x_i), 1 \leq i, j \leq M$.

The procedure is well posed if X is of sufficently large dimension (for $M \leq M_{max} \leq \dim X$).

Note that the matrix B^M is invertible and lower triangular (the diagonal is Id).

Approximation in a space of small n-width.

The recursion formula

We then define

$$\varphi_M = \arg \max_{\Phi \in \mathcal{S}_1[X]} \varepsilon_{M-1}(\Phi) \, , \tag{5}$$

and

$$\zeta_M = \arg \max_{x \in \Omega} \|\varphi_M(x) - \mathcal{I}_{M-1}[\varphi_M(x)])\|_{L^\infty(\Omega)} \, , \tag{6}$$

we finally set $q_M(x) = \frac{\varphi_M - \mathcal{I}_{M-1}[\varphi_M]}{\varphi_M(\zeta_M) - \mathcal{I}_{M-1}[\varphi_M](\zeta_M)}$ and $B^M_{ij} = q_j(x_i), 1 \leq i, j \leq M$.

The procedure is well posed if X is of sufficently large dimension (for $M \leq M_{max} \leq \dim X$).

Note that the matrix B^M is invertible and lower triangular (the diagonal is Id).

We then define

$$\varphi_M = \arg \max_{\Phi \in \mathcal{S}_1[X]} \varepsilon_{M-1}(\Phi) \ , \tag{5}$$

and

$$\zeta_M = \arg \max_{x \in \Omega} \|\varphi_M(x) - \mathcal{I}_{M-1}[\varphi_M(x)])\|_{L^\infty(\Omega)} \ , \tag{6}$$

we finally set $q_M(x) = \frac{\varphi_M - \mathcal{I}_{M-1}[\varphi_M]}{\varphi_M(\zeta_M) - \mathcal{I}_{M-1}[\varphi_M](\zeta_M)}$ and $B^M_{ij} = q_j(x_i), 1 \leq i, j \leq M$.

The procedure is well posed if X is of sufficently large dimension (for $M \leq M_{max} \leq \dim X$).

Note that the matrix B^M is invertible and lower triangular (the diagonal is Id).

We then define

$$\varphi_M = \arg \max_{\Phi \in \mathcal{S}_1[X]} \varepsilon_{M-1}(\Phi) \ , \tag{5}$$

and

$$\zeta_M = \arg \max_{x \in \Omega} \|\varphi_M(x) - \mathcal{I}_{M-1}[\varphi_M(x)])\|_{L^\infty(\Omega)} \ , \tag{6}$$

we finally set $q_M(x) = \frac{\varphi_M - \mathcal{I}_{M-1}[\varphi_M]}{\varphi_M(\zeta_M) - \mathcal{I}_{M-1}[\varphi_M](\zeta_M)}$ and $B^M_{ij} = q_j(x_i), 1 \leq i, j \leq M$.

The procedure is well posed if X is of sufficently large dimension (for $M \leq M_{max} \leq \dim X$).

Note that the matrix B^M is invertible and lower triangular (the diagonal is Id).

Approximation in a space of small n-width.

The Lebesgue constant

The error analysis of the interpolation procedure classically involves the Lebesgue constant $\Lambda_M = \sup_{x\in\Omega} \sum_{i=1}^{M} |h_i^M(x)|$, where the h_i^M is the associated Lagrange basis.

A (in practice very pessimistic) upper-bound for the Lebesgue constant is $2^M - 1$.

We remind also that the Lebesgue constant enters into the bound for the interpolation error as follows

Lemma

For any $u \in X$, the interpolation error satisfies

$$\|\Phi - \mathcal{I}_M\Phi\|_{L^\infty(\Omega)} \leq (1+\Lambda_M) \inf_{\psi_M \in span\{\varphi_i, 1\leq i\leq M\}} \|\Phi - \psi_M\|_{L^\infty(\Omega)}. \quad (7)$$

Approximation in a space of small n-width.

The Lebesgue constant

The error analysis of the interpolation procedure classically involves the Lebesgue constant $\Lambda_M = \sup_{x\in\Omega} \sum_{i=1}^{M} |h_i^M(x)|$, where the h_i^M is the associated Lagrange basis.

A (in practice very pessimistic) upper-bound for the Lebesgue constant is $2^M - 1$.

We remind also that the Lebesgue constant enters into the bound for the interpolation error as follows

Lemma

For any $u \in X$, the interpolation error satisfies

$$\|\Phi - \mathcal{I}_M\Phi\|_{L^\infty(\Omega)} \leq (1+\Lambda_M) \inf_{\psi_M \in span\{\varphi_i, 1\leq i\leq M\}} \|\Phi - \psi_M\|_{L^\infty(\Omega)}. \quad (7)$$

Approximation in a space of small n-width.

The Lebesgue constant

The error analysis of the interpolation procedure classically involves the Lebesgue constant $\Lambda_M = \sup_{x\in\Omega} \sum_{i=1}^{M} |h_i^M(x)|$, where the h_i^M is the associated Lagrange basis.
A (in practice very pessimistic) upper-bound for the Lebesgue constant is $2^M - 1$.
We remind also that the Lebesgue constant enters into the bound for the interpolation error as follows

Lemma

For any $u \in X$, the interpolation error satisfies

$$\|\Phi - \mathcal{I}_M\Phi\|_{L^\infty(\Omega)} \leq (1+\Lambda_M) \inf_{\psi_M \in span\{\varphi_i, 1\leq i\leq M\}} \|\Phi - \psi_M\|_{L^\infty(\Omega)}. \quad (7)$$

Approximation in a space of small n-width.

The Lebesgue constant

The error analysis of the interpolation procedure classically involves the Lebesgue constant $\Lambda_M = \sup_{x\in\Omega} \sum_{i=1}^{M} |h_i^M(x)|$, where the h_i^M is the associated Lagrange basis.
A (in practice very pessimistic) upper-bound for the Lebesgue constant is $2^M - 1$.
We remind also that the Lebesgue constant enters into the bound for the interpolation error as follows

Lemma

For any $u \in X$, the interpolation error satisfies

$$\|\Phi - \mathcal{I}_M\Phi\|_{L^\infty(\Omega)} \leq (1+\Lambda_M) \inf_{\psi_M \in \mathit{span}\{\varphi_i, 1\leq i\leq M\}} \|\Phi - \psi_M\|_{L^\infty(\Omega)}. \quad (7)$$

We can also prove that

Theorem

Assume that there exists a sequence of finite dimensional spaces

$$X_1 \subset X_2 \subset \cdots \subset X_M \subset \cdots \subset X, \quad \dim X_M = M \tag{8}$$

such that there exists $c > 0$ *and* α *with*

$$\forall \Phi \in X, \inf_{\psi_M \in X_M} \|\Phi - \psi_M\|_X \leq c e^{-\alpha M} \tag{9}$$

then, if $\alpha > \log(4)$, *there exists* $\beta > 0$ *such that*

$$\|\Phi - \mathcal{I}_M \Phi\|_{L^\infty(\Omega)} \leq c e^{-\beta M}. \tag{10}$$

Interpolation error..... a posteriori

For an estimator on the error, let $M \leq M_{\max} - 1$, we define

$$\hat{\varepsilon}_M(\Phi) \equiv |\Phi(\zeta_{M+1}) - \mathcal{I}_M\Phi(\zeta_{M+1})|$$

Lemma : If $\Phi \in X_{M+1}$, then

$$\|\Phi(\,\cdot) - \mathcal{I}_M\Phi(\,\cdot)\|_{L^\infty(\Omega)} \leq \hat{\varepsilon}_M(\Phi)$$

Of course, in general $\Phi \notin X_{M+1}$
and hence our estimator $\hat{\varepsilon}_M(\Phi)$ is not a rigorous upper bound;

however, if $\varepsilon_M(\Phi) \to 0$ very fast,

we expect (and check) that the effectivity, $\eta_M(\Phi) \equiv \hat{\varepsilon}_M(\Phi)/\varepsilon_M(\Phi) \simeq 1$.

Furthermore, the estimator is very inexpensive – *one additional evaluation* of Φ.

Interpolation error..... a posteriori

For an estimator on the error, let $M \leq M_{\max} - 1$, we define

$$\hat{\varepsilon}_M(\Phi) \equiv |\Phi(\zeta_{M+1}) - \mathcal{I}_M\Phi(\zeta_{M+1})|$$

Lemma : If $\Phi \in X_{M+1}$, then

$$\|\Phi(\,\cdot) - \mathcal{I}_M\Phi(\,\cdot)\|_{L^\infty(\Omega)} \leq \hat{\varepsilon}_M(\Phi)$$

Of course, in general $\Phi \notin X_{M+1}$
and hence our estimator $\hat{\varepsilon}_M(\Phi)$ is not a rigorous upper bound;

however, if $\varepsilon_M(\Phi) \to 0$ very fast,

we expect (and check) that the effectivity, $\eta_M(\Phi) \equiv \hat{\varepsilon}_M(\Phi)/\varepsilon_M(\Phi) \simeq 1$.

Furthermore, the estimator is very inexpensive – *one additional evaluation* of Φ.

Interpolation error..... a posteriori

For an estimator on the error, let $M \leq M_{\max} - 1$, we define

$$\hat{\varepsilon}_M(\Phi) \equiv |\Phi(\zeta_{M+1}) - \mathcal{I}_M\Phi(\zeta_{M+1})|$$

Lemma : If $\Phi \in X_{M+1}$, then

$$\|\Phi(\,\cdot) - \mathcal{I}_M\Phi(\,\cdot)\|_{L^\infty(\Omega)} \leq \hat{\varepsilon}_M(\Phi)$$

Of course, in general $\Phi \notin X_{M+1}$
and hence our estimator $\hat{\varepsilon}_M(\Phi)$ is not a rigorous upper bound;

however, if $\varepsilon_M(\Phi) \to 0$ very fast,

we expect (and check) that the effectivity, $\eta_M(\Phi) \equiv \hat{\varepsilon}_M(\Phi)/\varepsilon_M(\Phi) \simeq 1$.

Furthermore, the estimator is very inexpensive – *one additional evaluation* of Φ.

Interpolation error..... a posteriori

For an estimator on the error, let $M \leq M_{\max} - 1$, we define

$$\hat{\varepsilon}_M(\Phi) \equiv |\Phi(\zeta_{M+1}) - \mathcal{I}_M\Phi(\zeta_{M+1})|$$

Lemma : If $\Phi \in X_{M+1}$, then

$$\|\Phi(\,\cdot) - \mathcal{I}_M\Phi(\,\cdot)\|_{L^\infty(\Omega)} \leq \hat{\varepsilon}_M(\Phi)$$

Of course, in general $\Phi \notin X_{M+1}$
and hence our estimator $\hat{\varepsilon}_M(\Phi)$ is not a rigorous upper bound;

however, if $\varepsilon_M(\Phi) \to 0$ very fast,

we expect (and check) that the effectivity, $\eta_M(\Phi) \equiv \hat{\varepsilon}_M(\Phi)/\varepsilon_M(\Phi) \simeq 1$.

Furthermore, the estimator is very inexpensive – *one additional evaluation* of Φ.

Interpolation error..... a posteriori

For an estimator on the error, let $M \leq M_{\max} - 1$, we define

$$\hat{\varepsilon}_M(\Phi) \equiv |\Phi(\zeta_{M+1}) - \mathcal{I}_M\Phi(\zeta_{M+1})|$$

Lemma : If $\Phi \in X_{M+1}$, then

$$\|\Phi(\,\cdot) - \mathcal{I}_M\Phi(\,\cdot)\|_{L^\infty(\Omega)} \leq \hat{\varepsilon}_M(\Phi)$$

Of course, in general $\Phi \notin X_{M+1}$
and hence our estimator $\hat{\varepsilon}_M(\Phi)$ is not a rigorous upper bound;

however, if $\varepsilon_M(\Phi) \to 0$ very fast,

we expect (and check) that the effectivity, $\eta_M(\Phi) \equiv \hat{\varepsilon}_M(\Phi)/\varepsilon_M(\Phi) \simeq 1$.

Furthermore, the estimator is very inexpensive – *one additional evaluation* of Φ.

Interpolation error..... a posteriori

For an estimator on the error, let $M \leq M_{\max} - 1$, we define

$$\hat{\varepsilon}_M(\Phi) \equiv |\Phi(\zeta_{M+1}) - \mathcal{I}_M\Phi(\zeta_{M+1})|$$

Lemma : If $\Phi \in X_{M+1}$, then

$$\|\Phi(\,\cdot) - \mathcal{I}_M\Phi(\,\cdot)\|_{L^\infty(\Omega)} \leq \hat{\varepsilon}_M(\Phi)$$

Of course, in general $\Phi \notin X_{M+1}$
and hence our estimator $\hat{\varepsilon}_M(\Phi)$ is not a rigorous upper bound;

however, if $\varepsilon_M(\Phi) \to 0$ very fast,

we expect (and check) that the effectivity, $\eta_M(\Phi) \equiv \hat{\varepsilon}_M(\Phi)/\varepsilon_M(\Phi) \simeq 1$.

Furthermore, the estimator is very inexpensive – *one additional evaluation* of Φ.

Numerical results

We consider
$\Phi(x) \equiv \Phi\left((x_1, x_2); (\mu_1, \mu_2)\right) \equiv \left((x_1 - \mu_1)^2 + (x_2 - \mu_2)^2\right)^{-1/2}$ for $x \in]0,1[^2$ and $\mu \in [-1, -0.01]^2$

M	$\varepsilon^*_{M,\max}$	$\overline{\rho}_M$	Λ_M	$\overline{\eta}_M$
8	8.30 E-02	0.68	1.76	0.17
16	4.20 E-03	0.67	2.63	0.1.
24	2.68 E-04	0.49	4.42	0.28
32	5.64 E-05	0.48	5.15	0.20
40	3.66 E-06	0.54	4.98	0.60
48	6.08 E-07	0.37	7.43	0.29

$\varepsilon^*_{M,\max}$ is the best fit error, $\overline{\rho}_M$ is the averaged ratio $\dfrac{\varepsilon}{\varepsilon^*(1+\Lambda)}$, Λ_M is the "Lebesgue" constant and $\overline{\eta}_M$ is the averaged effectivity index $\dfrac{\hat{\varepsilon}}{\varepsilon}$

Numerical results

We consider
$\Phi(x) \equiv \Phi((x_1, x_2); (\mu_1, \mu_2)) \equiv \left((x_1 - \mu_1)^2 + (x_2 - \mu_2)^2\right)^{-1/2}$ for $x \in]0,1[^2$ and $\mu \in [-1, -0.01]^2$

M	$\varepsilon^*_{M,\max}$	$\overline{\rho}_M$	Λ_M	$\overline{\eta}_M$
8	8.30 E-02	0.68	1.76	0.17
16	4.20 E-03	0.67	2.63	0.1.
24	2.68 E-04	0.49	4.42	0.28
32	5.64 E-05	0.48	5.15	0.20
40	3.66 E-06	0.54	4.98	0.60
48	6.08 E-07	0.37	7.43	0.29

Note that we have here approximated the full set of $\Phi((x); \mu)$ with a few of them $\Phi((x); (\mu)) \simeq \sum_{i=1}^{M} \alpha_i \Phi\left((x); \mu^i\right)$

Numerical results

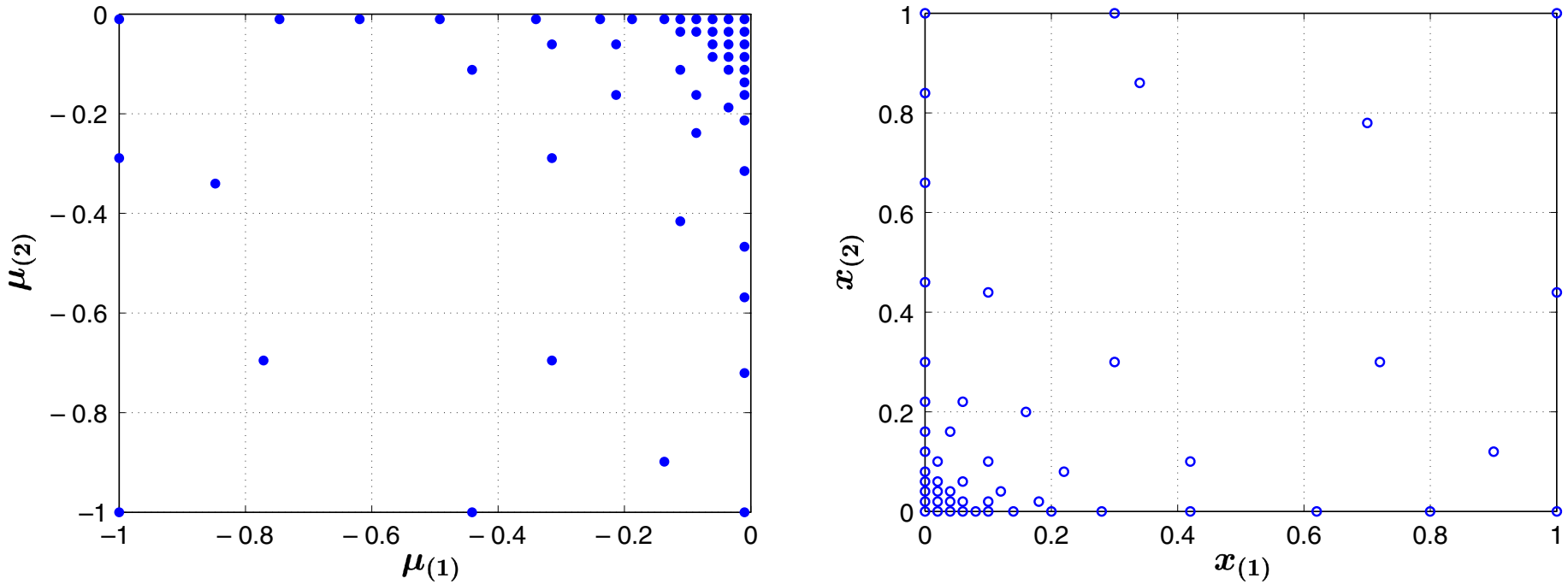


Figure: (a) Parameter sample set S_M^g, $M_{\max} = 51$, and (b) interpolation points x_m, $1 \leq m \leq M_{\max}$.

Outline

Application to polynomial interpolation

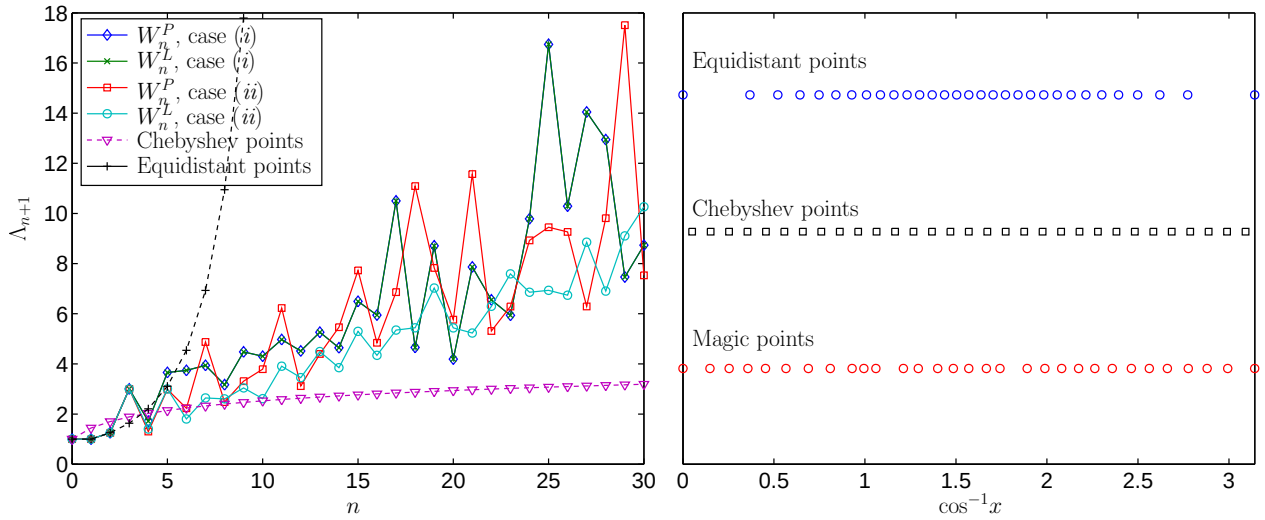


Figure: Lebesgue constant on the interval.

Application to polynomial interpolation

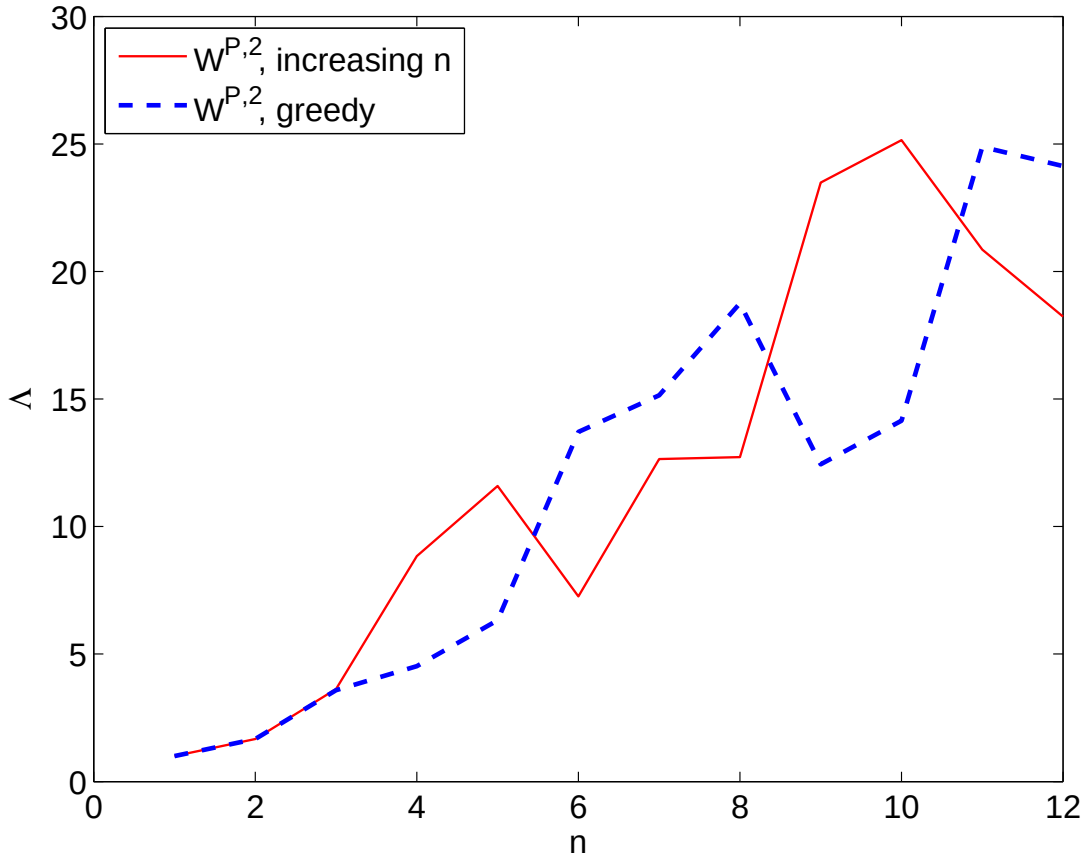


Figure: Lebesgue constant on the the triangle.

Application to polynomial interpolation

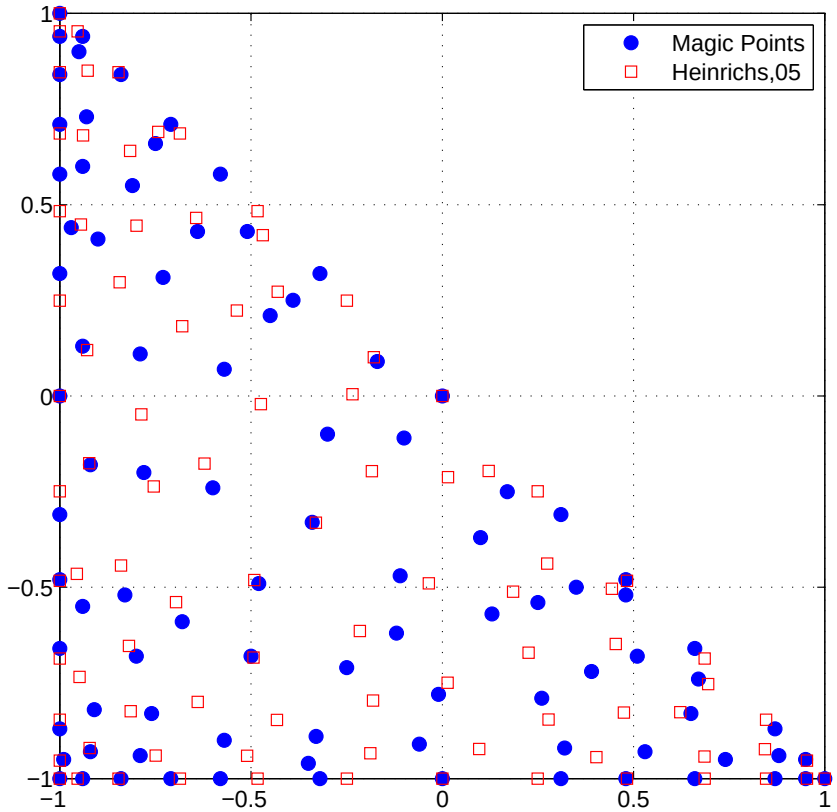


Figure: Disposition of the points on the the triangle.

Application to polynomial interpolation

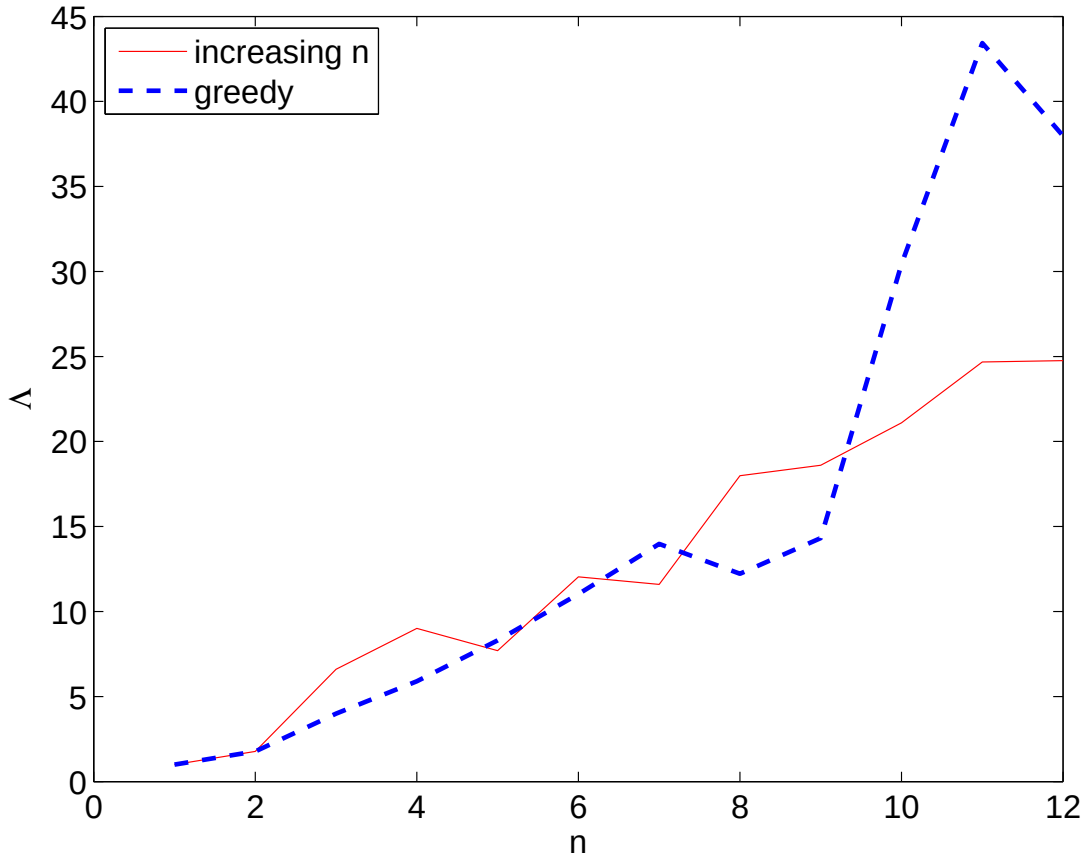


Figure: Lebesgue constant on the the pentagon.

Application to polynomial interpolation

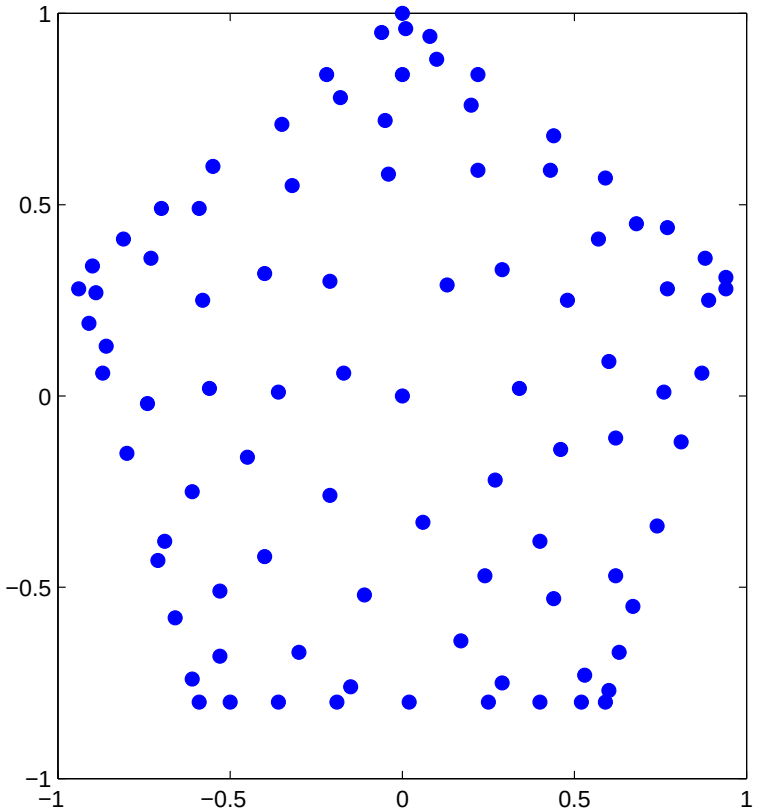


Figure: Disposition of the points on the the pentagon.

Application to polynomial interpolation

n	Magic Points	[12]	[6]
2	2.0	2.0	2.0
3	3.80	2.93	2.93
4	8.70	4.07	4.11
5	9.77	5.38	5.62
6	15.27	7.53	7.36
7	31.04	10.17	9.37
8	34.31	14.63	12.31
9	62.99	20.46	15.69

Figure: Lebesgue constant on the the tetrahedron.

Application to polynomial interpolation

Application to polynomial interpolation

n	M	$\lvert\varphi_{1d}(x_{M+1}) - \mathcal{I}_M[\varphi_{1d}(x_{M+1})]\rvert$	$\lVert\varphi_{1d} - \mathcal{I}_M\varphi_{1d}\rVert_{L^\infty}$	η_M
2	3	7.27 E−2	7.79 E−2	1.07
4	5	7.47 E−3	7.52 E−3	1.01
6	7	6.18 E−4	6.70 E−4	1.08
8	9	3.84 E−5	3.84 E−5	1.00
10	11	1.69 E−6	1.72 E−6	1.02
12	13	3.08 E−8	4.02 E−8	1.30
14	15	1.65 E−9	1.65 E−9	1.00
16	17	6.33 E−11	6.73 E−11	1.06
18	19	1.39 E−12	1.39 E−12	1.00
20	21	2.50 E−14	2.51 E−14	1.00

Table: Comparison between the error estimate and the actual error, for polynomial interpolation of $\varphi_{1d} = e^{-x^2}$.

Application to polynomial interpolation

n	M	$\lvert\varphi_{2d}(x_{M+1}) - \mathcal{I}_M[\varphi_{2d}(x_{M+1})]\rvert$	$\lVert\varphi_{2d} - \mathcal{I}_M\varphi_{2d}\rVert_{L^\infty}$	η_M
2	9	1.13 E−1	6.32 E−1	5.59
4	25	1.43 E−1	1.66 E−1	1.16
6	49	2.03 E−2	2.24 E−2	1.10
8	81	7.23 E−4	1.46 E−3	2.02
10	121	5.36 E−5	1.06 E−4	1.98
12	169	2.76 E−6	2.78 E−6	1.01
14	225	1.04 E−8	1.31 E−7	12.60
16	289	2.67 E−9	4.88 E−9	1.83
18	361	4.98 E−11	1.16 E−10	2.33
20	441	2.57 E−12	2.78 E−12	1.08

Table: Comparison between the error estimate and the actual error, for polynomial interpolation of $\varphi_{2d} = e^{-(x^2+y^2)}$.

Application to polynomial interpolation

n	M	$\lvert\varphi_{\mathrm{irr}}(x_{M+1}) - \mathcal{I}_M[\varphi_{\mathrm{irr}}(x_{M+1})]\rvert$	$\lVert\varphi_{\mathrm{irr}} - \mathcal{I}_M\varphi_{\mathrm{irr}}\rVert_{L^\infty}$	η_M
2	9	$7.95\,\mathrm{E}{-2}$	$1.59\,\mathrm{E}{-1}$	2.00
4	25	$3.88\,\mathrm{E}{-2}$	$1.47\,\mathrm{E}{-1}$	3.79
6	49	$2.44\,\mathrm{E}{-3}$	$1.95\,\mathrm{E}{-2}$	8.00
8	81	$4.26\,\mathrm{E}{-3}$	$2.42\,\mathrm{E}{-2}$	5.68
10	121	$1.37\,\mathrm{E}{-3}$	$3.74\,\mathrm{E}{-3}$	2.73
12	169	$3.75\,\mathrm{E}{-3}$	$5.66\,\mathrm{E}{-3}$	1.51
14	225	$2.96\,\mathrm{E}{-4}$	$5.69\,\mathrm{E}{-4}$	1.92
16	289	$5.01\,\mathrm{E}{-5}$	$5.80\,\mathrm{E}{-4}$	11.58
18	361	$1.29\,\mathrm{E}{-4}$	$3.00\,\mathrm{E}{-4}$	2.33
20	441	$3.09\,\mathrm{E}{-4}$	$5.72\,\mathrm{E}{-4}$	1.85

Table: Comparison between the error estimate and the actual error, for polynomial interpolation of $\varphi_{\mathrm{irr}} = |x^3y^3|$.

Outline

Reduced Basis Method : Framework of the approach.

Basics on approximation

- A lot of problems we have to face in numerical analysis and scientific computing: find u such that

$$\mathcal{F}(u) = 0 \tag{1}$$

can actually be written under a variational form : find $u \in \mathcal{X}$ such that

$$\mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2}$$

Where $\mathcal{X}$ and $\tilde{\mathcal{X}}$ are some coherent Banach spaces, $\mathcal{A}$ is an appropriate continuous form, linear in v, and f is a given linear form.
For time dependent problem one may specify even : find $u, \forall t, u(t, ;) \in \mathcal{X}$ such that

$$m(\frac{\partial u}{\partial t}, v) + \mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2'}$$

Reduced Basis Method : Framework of the approach.

Basics on approximation

- A lot of problems we have to face in numerical analysis and scientific computing: find u such that

$$\mathcal{F}(u) = 0 \tag{1}$$

can actually be written under a variational form : find $u \in \mathcal{X}$ such that

$$\mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2}$$

Where $\mathcal{X}$ and $\tilde{\mathcal{X}}$ are some coherent Banach spaces, $\mathcal{A}$ is an appropriate continuous form, linear in v, and f is a given linear form.
For time dependent problem one may specify even : find $u, \forall t, u(t, ;) \in \mathcal{X}$ such that

$$m(\frac{\partial u}{\partial t}, v) + \mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2'}$$

Reduced Basis Method : Framework of the approach.

Basics on approximation

- A lot of problems we have to face in numerical analysis and scientific computing: find u such that

$$\mathcal{F}(u) = 0 \tag{1}$$

  can actually be written under a variational form : find $u \in \mathcal{X}$ such that

$$\mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2}$$

  Where $\mathcal{X}$ and $\tilde{\mathcal{X}}$ are some coherent Banach spaces, $\mathcal{A}$ is an appropriate continuous form, linear in v, and f is a given linear form.

  For time dependent problem one may specify even : find $u, \forall t, u(t, ;) \in \mathcal{X}$ such that

$$m(\frac{\partial u}{\partial t}, v) + \mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2'}$$

Reduced Basis Method : Framework of the approach.

Basics on approximation

- A lot of problems we have to face in numerical analysis and scientific computing: find u such that

$$\mathcal{F}(u) = 0 \tag{1}$$

can actually be written under a variational form : find $u \in \mathcal{X}$ such that

$$\mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2}$$

Where $\mathcal{X}$ and $\tilde{\mathcal{X}}$ are some coherent Banach spaces, $\mathcal{A}$ is an appropriate continuous form, linear in v, and f is a given linear form.
For time dependent problem one may specify even : find $u, \forall t, u(t, ;) \in \mathcal{X}$ such that

$$m(\frac{\partial u}{\partial t}, v) + \mathcal{A}(u, v) =< f, v >, \quad \forall v \in \tilde{\mathcal{X}} \tag{2'}$$

Reduced Basis Method : Framework of the approach.

Basics on approximation

The coherence in $\mathcal{X}$ and $\tilde{\mathcal{X}}$ is expressed through a condition in terms of $\mathcal{A}$ that, for linear problems, involves, e.g.

- the ellipticity or coercivity (Lax Milgram theorem when $\mathcal{X} = \tilde{\mathcal{X}}$) or
- the Babuška-Brezzi condition.....

that makes explicit conditions under which the problem is well posed : i.e. there exists a unique solution u to problem (1).

For nonlinear problems the conditions are various and more involved.

Reduced Basis Method : Framework of the approach.

Basics on approximation

The coherence in $\mathcal{X}$ and $\tilde{\mathcal{X}}$ is expressed through a condition in terms of $\mathcal{A}$ that, for linear problems, involves, e.g.

- the ellipticity or coercivity (Lax Milgram theorem when $\mathcal{X} = \tilde{\mathcal{X}}$) or
- the Babuška-Brezzi condition.....

that makes explicit conditions under which the problem is well posed : i.e. there exists a unique solution u to problem (1).

For nonlinear problems the conditions are various and more involved.

Reduced Basis Method : Framework of the approach.

Basics on approximation

The coherence in $\mathcal{X}$ and $\tilde{\mathcal{X}}$ is expressed through a condition in terms of $\mathcal{A}$ that, for linear problems, involves, e.g.

- the ellipticity or coercivity (Lax Milgram theorem when $\mathcal{X} = \tilde{\mathcal{X}}$) or
- the Babuška-Brezzi condition.....

that makes explicit conditions under which the problem is well posed : i.e. there exists a unique solution u to problem (1).

For nonlinear problems the conditions are various and more involved.

Reduced Basis Method : Framework of the approach.

Basics on approximation

The coherence in $\mathcal{X}$ and $\tilde{\mathcal{X}}$ is expressed through a condition in terms of $\mathcal{A}$ that, for linear problems, involves, e.g.

- the ellipticity or coercivity (Lax Milgram theorem when $\mathcal{X} = \tilde{\mathcal{X}}$) or
- the Babuška-Brezzi condition.....

that makes explicit conditions under which the problem is well posed : i.e. there exists a unique solution u to problem (1).

For nonlinear problems the conditions are various and more involved.

The coherence in $\mathcal{X}$ and $\tilde{\mathcal{X}}$ is expressed through a condition in terms of $\mathcal{A}$ that, for linear problems, involves, e.g.

- the ellipticity or coercivity (Lax Milgram theorem when $\mathcal{X} = \tilde{\mathcal{X}}$) or
- the Babuška-Brezzi condition.....

that makes explicit conditions under which the problem is well posed : i.e. there exists a unique solution *u* to problem (1).

For nonlinear problems the conditions are various and more involved.

The coherence in $\mathcal{X}$ and $\tilde{\mathcal{X}}$ is expressed through a condition in terms of $\mathcal{A}$ that, for linear problems, involves, e.g.

- the ellipticity or coercivity (Lax Milgram theorem when $\mathcal{X} = \tilde{\mathcal{X}}$) or
- the Babuška-Brezzi condition.....

that makes explicit conditions under which the problem is well posed : i.e. there exists a unique solution *u* to problem (1).

For nonlinear problems the conditions are various and more involved.

Reduced Basis Method : Framework of the approach.

Basics on approximation

The approximation can now proceed. Two families of finite dimensional spaces $\{\mathcal{X}_n\}_n$ and $\{\tilde{\mathcal{X}}_n\}_n$ are provided, that maintain the above mentioned coherence.

and the discrete space reads : find $u_n \in \mathcal{X}_n$ such that

$$\mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2_n)$$

or again for time dependent problems : find $u_n, \forall t, u_n(t,;) \in \mathcal{X}_n$ such that

$$m_n(\frac{\partial u_n}{\partial t}, v_n) + \mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2'_n)$$

most often further numerical quadratures are involved leading to slightly modified discrete problems

Reduced Basis Method : Framework of the approach.

Basics on approximation

The approximation can now proceed. Two families of finite dimensional spaces $\{\mathcal{X}_n\}_n$ and $\{\tilde{\mathcal{X}}_n\}_n$ are provided, that maintain the above mentioned coherence.
and the discrete space reads : find $u_n \in \mathcal{X}_n$ such that

$$\mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2_n)$$

or again for time dependent problems : find $u_n, \forall t, u_n(t,;) \in \mathcal{X}_n$ such that

$$m_n(\frac{\partial u_n}{\partial t}, v_n) + \mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2'_n)$$

most often further numerical quadratures are involved leading to slightly modified discrete problems

The approximation can now proceed. Two families of finite dimensional spaces $\{\mathcal{X}_n\}_n$ and $\{\tilde{\mathcal{X}}_n\}_n$ are provided, that maintain the above mentioned coherence.
and the discrete space reads : find $u_n \in \mathcal{X}_n$ such that

$$\mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2_n)$$

or again for time dependent problems : find $u_n, \forall t, u_n(t,;) \in \mathcal{X}_n$ such that

$$m_n(\frac{\partial u_n}{\partial t}, v_n) + \mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2'_n)$$

most often further numerical quadratures are involved leading to slightly modified discrete problems

The approximation can now proceed. Two families of finite dimensional spaces $\{\mathcal{X}_n\}_n$ and $\{\tilde{\mathcal{X}}_n\}_n$ are provided, that maintain the above mentioned coherence.
and the discrete space reads : find $u_n \in \mathcal{X}_n$ such that

$$\mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2_n)$$

or again for time dependent problems : find $u_n, \forall t, u_n(t,;) \in \mathcal{X}_n$ such that

$$m_n(\frac{\partial u_n}{\partial t}, v_n) + \mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2'_n)$$

most often further numerical quadratures are involved leading to slightly modified discrete problems

The approximation can now proceed. Two families of finite dimensional spaces $\{\mathcal{X}_n\}_n$ and $\{\tilde{\mathcal{X}}_n\}_n$ are provided, that maintain the above mentioned coherence.
and the discrete space reads : find $u_n \in \mathcal{X}_n$ such that

$$\mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2_n)$$

or again for time dependent problems : find $u_n, \forall t, u_n(t,;) \in \mathcal{X}_n$ such that

$$m_n(\frac{\partial u_n}{\partial t}, v_n) + \mathcal{A}_n(u_n, v_n) =< f_n, v_n >, \quad \forall v_n \in \tilde{\mathcal{X}}_n \qquad (2'_n)$$

most often further numerical quadratures are involved leading to slightly modified discrete problems

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

Basics on approximation

All the art of the numerical analyst or the specialist of scientific computing tends to define the discrete spaces $\mathcal{X}_n$ and $\tilde{\mathcal{X}}_n$ ... and also $\mathcal{A}_n$.... in such a way that

- The discrete solutions u_n exist and are unique
- An error bound $\|u - u_N\|_{\mathcal{X}} \leq c \inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$ can be derived
- The best fit, $\inf_{w_n \in \mathcal{X}_n} \|u - w_N\|_{\mathcal{X}}$, goes to zero rapidly
- The effective computation of u_n is easy enough
- An a posteriori error providing the size of $\|u - u_N\|_{\mathcal{X}}$ is available
- An a posteriori indicator telling what to do to improve $\|u - u_N\|_{\mathcal{X}}$ is available

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What most approaches do

Most approaches tend to define a family of spaces $\mathcal{X}_n$ either by proposing

- multipurpose approximations providing good accuracy assuming some regularity holds this is the case of finite differences, finite element, finite volume, spectral... methods

then error in lower order spaces (Aubin's trick)

- nonlinear approximations based on a posteriori indicators that allows refinements or based on multiresolution analysis

Reduced Basis Method : Framework of the approach.

What reduced basis method propose

Here what we want to use is an a priori knowledge of a reduced space X much smaller than $\mathcal{X}$ where the solution to (1) should be sought

We present here two classes of problems where this strategy can be used

- parameter dependent problems
- hierarchical geometry for the domain

in both cases the space X is conceived from the use of a more standard approximation methods you do not have to forget your favorite method... it is more the opposite in a first step

Reduced Basis Method : Framework of the approach.

What reduced basis method propose

Here what we want to use is an a priori knowledge of a reduced space X much smaller than $\mathcal{X}$ where the solution to (1) should be sought
We present here two classes of problems where this strategy can be used

- parameter dependent problems
- hierarchical geometry for the domain

in both cases the space X is conceived from the use of a more standard approximation methods you do not have to forget your favorite method... it is more the opposite in a first step

Reduced Basis Method : Framework of the approach.

What reduced basis method propose

Here what we want to use is an a priori knowledge of a reduced space X much smaller than $\mathcal{X}$ where the solution to (1) should be sought
We present here two classes of problems where this strategy can be used

- parameter dependent problems
- hierarchical geometry for the domain

in both cases the space X is conceived from the use of a more standard approximation methods you do not have to forget your favorite method... it is more the opposite in a first step

Reduced Basis Method : Framework of the approach.

What reduced basis method propose

Here what we want to use is an a priori knowledge of a reduced space X much smaller than $\mathcal{X}$ where the solution to (1) should be sought
We present here two classes of problems where this strategy can be used

- parameter dependent problems
- hierarchical geometry for the domain

in both cases the space X is conceived from the use of a more standard approximation methods you do not have to forget your favorite method... it is more the opposite in a first step

Reduced Basis Method : Framework of the approach.

What reduced basis method propose

Here what we want to use is an a priori knowledge of a reduced space X much smaller than $\mathcal{X}$ where the solution to (1) should be sought
We present here two classes of problems where this strategy can be used

- parameter dependent problems
- hierarchical geometry for the domain

in both cases the space X is conceived from the use of a more standard approximation methods you do not have to forget your favorite method... it is more the opposite in a first step

Reduced Basis Method : Framework of the approach.

What reduced basis method propose

Here what we want to use is an a priori knowledge of a reduced space X much smaller than $\mathcal{X}$ where the solution to (1) should be sought
We present here two classes of problems where this strategy can be used

- parameter dependent problems
- hierarchical geometry for the domain

in both cases the space X is conceived from the use of a more standard approximation methods you do not have to forget your favorite method... it is more the opposite in a first step

Outline

Parameter dependent problems.

Basics

Let us consider a class of problems depending on some parameters:

$$\mathcal{F}(u, \mu) = 0 \qquad (1')$$

and the parameter μ belongs to R^d (or some brick in R^d)

- This is the case for instance in a dimensional problem where some parameters have to be optimized for some purpose
- This can equally be the case for an inverse problem in parameter identification.
- The solution $u = u(\mu)$ of (1') is sought in some space $\mathcal{X}$ for any given parameter μ
- The dependancy in μ of the solution $u(\mu)$ is most often regular.

Parameter dependent problems.

Basics

Let us consider a class of problems depending on some parameters:

$$\mathcal{F}(u, \mu) = 0 \qquad (1')$$

and the parameter μ belongs to R^d (or some brick in R^d)

- This is the case for instance in a dimensional problem where some parameters have to be optimized for some purpose
- This can equally be the case for an inverse problem in parameter identification.
- The solution $u = u(\mu)$ of (1') is sought in some space $\mathcal{X}$ for any given parameter μ
- The dependancy in μ of the solution $u(\mu)$ is most often regular.

Parameter dependent problems.

Basics

Let us consider a class of problems depending on some parameters:

$$\mathcal{F}(u, \mu) = 0 \qquad (1')$$

and the parameter μ belongs to R^d (or some brick in R^d)

- This is the case for instance in a dimensional problem where some parameters have to be optimized for some purpose
- This can equally be the case for an inverse problem in parameter identification.
- The solution $u = u(\mu)$ of (1') is sought in some space $\mathcal{X}$ for any given parameter μ
- The dependancy in μ of the solution $u(\mu)$ is most often regular.

Parameter dependent problems.

Basics

Let us consider a class of problems depending on some parameters:

$$\mathcal{F}(u, \mu) = 0 \qquad (1')$$

and the parameter μ belongs to R^d (or some brick in R^d)

- This is the case for instance in a dimensional problem where some parameters have to be optimized for some purpose
- This can equally be the case for an inverse problem in parameter identification.
- The solution $u = u(\mu)$ of (1') is sought in some space $\mathcal{X}$ for any given parameter μ
- The dependancy in μ of the solution $u(\mu)$ is most often regular.

Parameter dependent problems.

Basics

Let us consider a class of problems depending on some parameters:

$$\mathcal{F}(u, \mu) = 0 \qquad (1')$$

and the parameter μ belongs to R^d (or some brick in R^d)

- This is the case for instance in a dimensional problem where some parameters have to be optimized for some purpose
- This can equally be the case for an inverse problem in parameter identification.
- The solution $u = u(\mu)$ of (1') is sought in some space $\mathcal{X}$ for any given parameter μ
- The dependancy in μ of the solution $u(\mu)$ is most often regular.

Parameter dependent problems.

Basics

Let us consider a class of problems depending on some parameters:

$$\mathcal{F}(u, \mu) = 0 \qquad (1')$$

and the parameter μ belongs to R^d (or some brick in R^d)

- This is the case for instance in a dimensional problem where some parameters have to be optimized for some purpose
- This can equally be the case for an inverse problem in parameter identification.
- The solution $u = u(\mu)$ of (1') is sought in some space $\mathcal{X}$ for any given parameter μ
- The dependancy in μ of the solution $u(\mu)$ is most often regular.

Parameter dependent problems.

The reduced basis space and approximation

- Define $X = \text{Span}\{u(\mu),\ \mu \in \mathcal{D}\}$ then looking for the solution in X instead of $\mathcal{X}$ (generally a Sobolev space) is already a valuable indication.....
- In order to apprehend in which sense the good behavior of X should be understood, it is helpfull to introduce the notion of n-width following Kolmogorov

Parameter dependent problems.

The reduced basis space and approximation

- Define $X = \text{Span}\{u(\mu),\ \mu \in \mathcal{D}\}$ then looking for the solution in X instead of $\mathcal{X}$ (generally a Sobolev space) is already a valuable indication.....
- In order to apprehend in which sense the good behavior of X should be understood, it is helpfull to introduce the notion of n-width following Kolmogorov

Definition

Let $\mathcal{X}$ be a normed linear space, X be a subset of $\mathcal{X}$ and X_n be a generic n-dimensional subspace of $\mathcal{X}$. The deviation of X from X_n is

$$E(X; X_n) = \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}}.$$

The *Kolmogorov n-width* of X in X is given by

$$\begin{aligned} d_n(X, \mathcal{X}) &= \inf\{E(X; X_n) : X_n \text{ an } n\text{-dimensional subspace of X}\} \\ &= \inf_{X_n} \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}} \ . \end{aligned} \tag{11}$$

The n-width of X thus measures the extent to which X may be approximated by a n-dimensional subspace of $\mathcal{X}$.

Definition

Let $\mathcal{X}$ be a normed linear space, X be a subset of $\mathcal{X}$ and X_n be a generic n-dimensional subspace of $\mathcal{X}$. The deviation of X from X_n is

$$E(X; X_n) = \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}}.$$

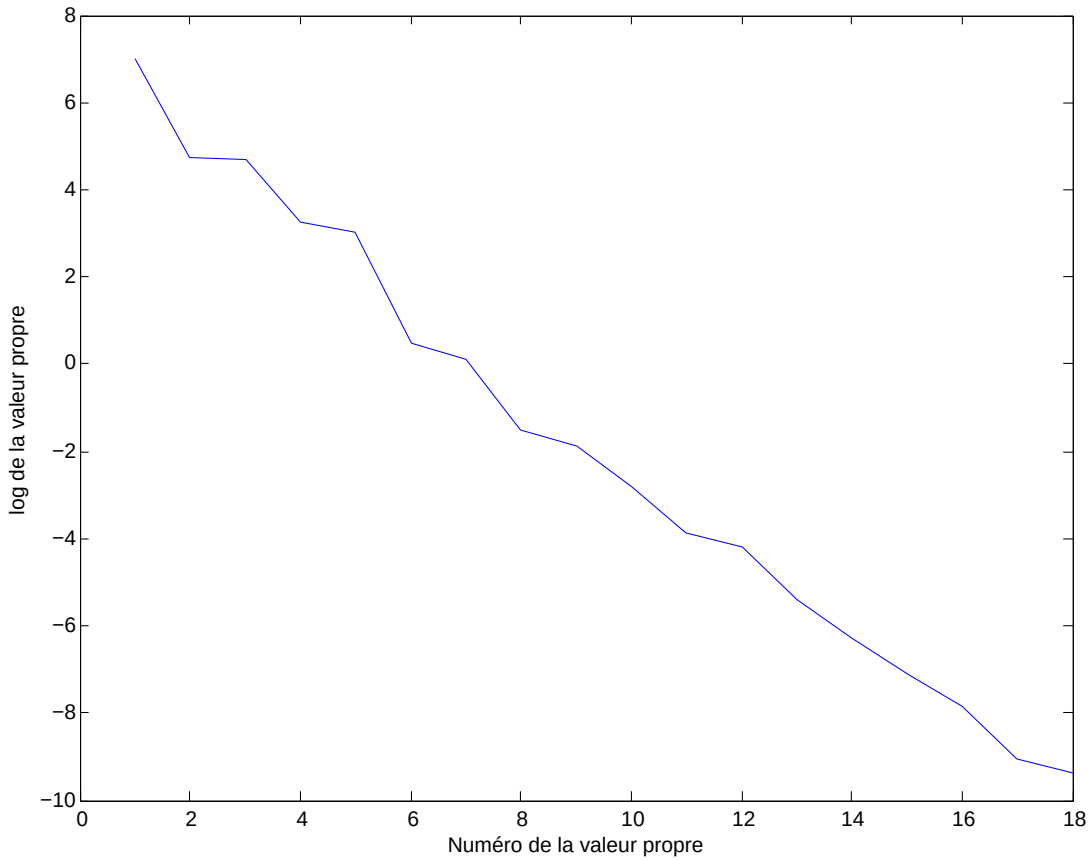
The *Kolmogorov n-width* of X in X is given by

$$\begin{aligned} d_n(X, \mathcal{X}) &= \inf\{E(X; X_n) : X_n \text{ an } n\text{-dimensional subspace of X}\} \\ &= \inf_{X_n} \sup_{x \in X} \inf_{y \in X_n} \|x - y\|_{\mathcal{X}} \ . \end{aligned} \tag{11}$$

The n-width of X thus measures the extent to which X may be approximated by a n-dimensional subspace of $\mathcal{X}$.

Evaluation of the n-width of the set of solutions.

- PCA in appropriate norms



Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Parameter dependent problems.

The reduced basis space and approximation

- Of course X is rarely known but $X_n = \text{Span}\{u(\mu_k),\ k = 1, ..., n\}$ where μ_k are properly chosen
- The solution to (1') for other values of μ is then approximated through a Galerkin process.
- The best fit approximation is often exponential in n and a random log repartition of the sample values μ_k is often better than other obvious choices.
  - Almroth B.O., Stern P., Brogan F.A.(1978)
  - Noor A.K., Peters J.M.(1980)
- Galerkin approximation is preferable to any kind of interpolation or extrapolation method.

Outline

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

An example

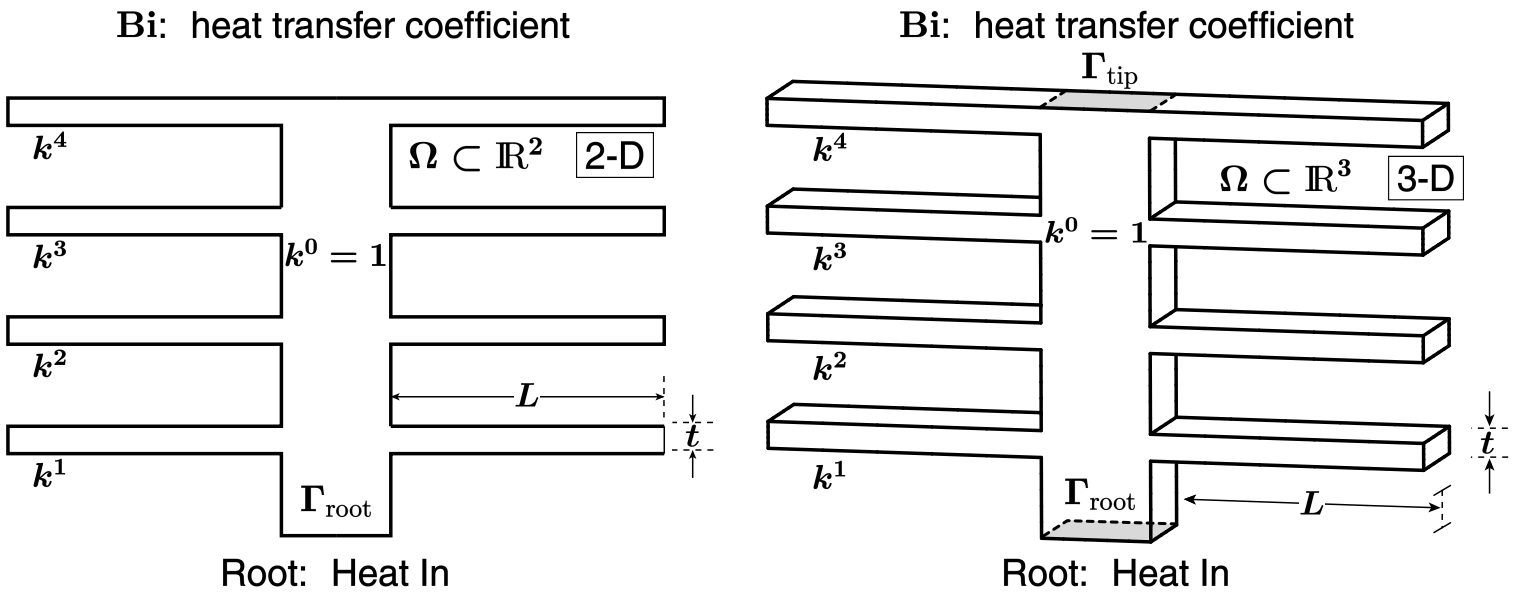


Figure: the fin geometry.

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

- the conductivities $k_i, i = 1, .., 4$
- the dimensions L and t
- the Biot number Bi
- The design space is $\mathcal{D} = [0.1, 10]^4 \times [0.01, 1] \times [2, 3] \times [0.1, 0.5]$

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

- the conductivities $k_i, i = 1, .., 4$
- the dimensions L and t
- the Biot number Bi
- The design space is $\mathcal{D} = [0.1, 10]^4 \times [0.01, 1] \times [2, 3] \times [0.1, 0.5]$

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

- the conductivities $k_i, i = 1, .., 4$
- the dimensions L and t
- the Biot number Bi
- The design space is $\mathcal{D} = [0.1, 10]^4 \times [0.01, 1] \times [2, 3] \times [0.1, 0.5]$

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

- the conductivities $k_i, i = 1, .., 4$
- the dimensions L and t
- the Biot number Bi
- The design space is $\mathcal{D} = [0.1, 10]^4 \times [0.01, 1] \times [2, 3] \times [0.1, 0.5]$

An example

Thermal fin problem

$$a(u, v; \mu) := \int_\Omega k \nabla u \nabla v + \mathrm{Bi} \int_{\partial\Omega \setminus \Gamma_1} uv = \int_{\Gamma_1} v$$

The parameters are :

- the conductivities $k_i, i = 1, .., 4$
- the dimensions L and t
- the Biot number Bi
- The design space is $\mathcal{D} = [0.1, 10]^4 \times [0.01, 1] \times [2, 3] \times [0.1, 0.5]$

An example

... and its approximation

We choose randomly N points in the design space $\mathcal{D}$
and compute the solutions for these points by a finite element method (say).
The reduced basis method is then implemented, from these solutions, and the error on the averaged temperature at the foot of the fin is

N	10	20	30	40	50
Error	1.610^{-1}	1.610^{-2}	2.410^{-3}	7.210^{-4}	3.110^{-4}

Note that even for $N = 50$ there are less than 2 points per parameter direction, exponential convergence

N appears rather not much dependent on the number of parameters, actually $N(d) << N(1)^d$!!!

An example

... and its approximation

We choose randomly N points in the design space $\mathcal{D}$
and compute the solutions for these points by a finite element method (say).

The reduced basis method is then implemented, from these solutions, and the error on the averaged temperature at the foot of the fin is

N	10	20	30	40	50
Error	1.610^{-1}	1.610^{-2}	2.410^{-3}	7.210^{-4}	3.110^{-4}

Note that even for $N = 50$ there are less than 2 points per parameter direction, exponential convergence

N appears rather not much dependent on the number of parameters, actually $N(d) << N(1)^d$!!!

An example

... and its approximation

We choose randomly N points in the design space $\mathcal{D}$
and compute the solutions for these points by a finite element method (say).
The reduced basis method is then implemented, from these solutions, and the error on the averaged temperature at the foot of the fin is

N	10	20	30	40	50
Error	1.610^{-1}	1.610^{-2}	2.410^{-3}	7.210^{-4}	3.110^{-4}

Note that even for $N = 50$ there are less than 2 points per parameter direction, exponential convergence

N appears rather not much dependent on the number of parameters, actually $N(d) << N(1)^d$!!!

An example

... and its approximation

We choose randomly N points in the design space $\mathcal{D}$
and compute the solutions for these points by a finite element method (say).
The reduced basis method is then implemented, from these solutions, and the error on the averaged temperature at the foot of the fin is

N	10	20	30	40	50
Error	1.610^{-1}	1.610^{-2}	2.410^{-3}	7.210^{-4}	3.110^{-4}

Note that even for $N = 50$ there are less than 2 points per parameter direction, exponential convergence

N appears rather not much dependent on the number of parameters, actually $N(d) << N(1)^d$!!!

An example

... and its approximation

We choose randomly N points in the design space $\mathcal{D}$
and compute the solutions for these points by a finite element method (say).
The reduced basis method is then implemented, from these solutions, and the error on the averaged temperature at the foot of the fin is

N	10	20	30	40	50
Error	1.610^{-1}	1.610^{-2}	2.410^{-3}	7.210^{-4}	3.110^{-4}

Note that even for $N = 50$ there are less than 2 points per parameter direction, exponential convergence

N appears rather not much dependent on the number of parameters, actually $N(d) << N(1)^d$!!!

An example

... and its approximation

We choose randomly N points in the design space $\mathcal{D}$
and compute the solutions for these points by a finite element method (say).
The reduced basis method is then implemented, from these solutions, and the error on the averaged temperature at the foot of the fin is

N	10	20	30	40	50
Error	1.610^{-1}	1.610^{-2}	2.410^{-3}	7.210^{-4}	3.110^{-4}

Note that even for $N = 50$ there are less than 2 points per parameter direction, exponential convergence

N appears rather not much dependent on the number of parameters, actually $N(d) << N(1)^d$!!!

Outline

Black-Box implementation

In order to solve the reduced basis problems in real time, some preliminary computations — off line — have to be made :

- e.g. the stiffness matrix $\mathcal{A}_{i,j} := a(u(\mu_i), u(\mu_j); \mu)$.

Note that $\Omega = \cup\Omega_\ell$ hence : $\int_\Omega = \sum_{\ell=0}^{6} \int_{\Omega_\ell}$

Note also that through a change of variable, each integral over the Ω_ℓ can be written as a linear combinaison of $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial x}\frac{\partial v}{\partial x}$ and $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial y}\frac{\partial v}{\partial y}$

so that the original problem

$$\int_\Omega k\nabla u\nabla v + \mathrm{Bi}\int_{\partial\Omega\setminus\Gamma_1} uv = \int_{\Gamma_1} v$$

can be rewritten as

$$\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$$

- where the bilinear forms a_p are parameter independent

Black-Box implementation

In order to solve the reduced basis problems in real time, some preliminary computations — off line — have to be made :

- e.g. the stiffness matrix $\mathcal{A}_{i,j} := a(u(\mu_i), u(\mu_j); \mu)$.

Note that $\Omega = \cup\Omega_\ell$ hence : $\int_\Omega = \sum_{\ell=0}^{6} \int_{\Omega_\ell}$

Note also that through a change of variable, each integral over the Ω_ℓ can be written as a linear combinaison of $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial x}\frac{\partial v}{\partial x}$ and $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial y}\frac{\partial v}{\partial y}$

so that the original problem

$$\int_\Omega k\nabla u\nabla v + \mathrm{Bi}\int_{\partial\Omega\setminus\Gamma_1} uv = \int_{\Gamma_1} v$$

can be rewritten as

$$\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$$

- where the bilinear forms a_p are parameter independent

Black-Box implementation

In order to solve the reduced basis problems in real time, some preliminary computations — off line — have to be made :

- e.g. the stiffness matrix $\mathcal{A}_{i,j} := a(u(\mu_i), u(\mu_j); \mu)$.

Note that $\Omega = \cup\Omega_\ell$ hence : $\int_\Omega = \sum_{\ell=0}^{6} \int_{\Omega_\ell}$

Note also that through a change of variable, each integral over the Ω_ℓ can be written as a linear combinaison of $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial x}\frac{\partial v}{\partial x}$ and $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial y}\frac{\partial v}{\partial y}$
so that the original problem

$$\int_\Omega k\nabla u\nabla v + \mathrm{Bi}\int_{\partial\Omega\setminus\Gamma_1} uv = \int_{\Gamma_1} v$$

can be rewritten as

$$\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$$

- where the bilinear forms a_p are parameter independent

Black-Box implementation

In order to solve the reduced basis problems in real time, some preliminary computations — off line — have to be made :

- e.g. the stiffness matrix $\mathcal{A}_{i,j} := a(u(\mu_i), u(\mu_j); \mu)$.

Note that $\Omega = \cup\Omega_\ell$ hence : $\int_\Omega = \sum_{\ell=0}^{6} \int_{\Omega_\ell}$

Note also that through a change of variable, each integral over the Ω_ℓ can be written as a linear combinaison of $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial x}\frac{\partial v}{\partial x}$ and $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial y}\frac{\partial v}{\partial y}$

so that the original problem

$$\int_\Omega k\nabla u\nabla v + \mathrm{Bi}\int_{\partial\Omega\setminus\Gamma_1} uv = \int_{\Gamma_1} v$$

can be rewritten as

$$\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$$

- where the bilinear forms a_p are parameter independent

Black-Box implementation

In order to solve the reduced basis problems in real time, some preliminary computations — off line — have to be made :

- e.g. the stiffness matrix $\mathcal{A}_{i,j} := a(u(\mu_i), u(\mu_j); \mu)$.

Note that $\Omega = \cup\Omega_\ell$ hence : $\int_\Omega = \sum_{\ell=0}^{6} \int_{\Omega_\ell}$

Note also that through a change of variable, each integral over the Ω_ℓ can be written as a linear combinaison of $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial x}\frac{\partial v}{\partial x}$ and $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial y}\frac{\partial v}{\partial y}$

so that the original problem

$$\int_\Omega k\nabla u\nabla v + \mathrm{Bi}\int_{\partial\Omega\setminus\Gamma_1} uv = \int_{\Gamma_1} v$$

can be rewritten as

$$\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$$

- where the bilinear forms a_p are parameter independent

Black-Box implementation

In order to solve the reduced basis problems in real time, some preliminary computations — off line — have to be made :

- e.g. the stiffness matrix $\mathcal{A}_{i,j} := a(u(\mu_i), u(\mu_j); \mu)$.

Note that $\Omega = \cup\Omega_\ell$ hence : $\int_\Omega = \sum_{\ell=0}^{6} \int_{\Omega_\ell}$

Note also that through a change of variable, each integral over the Ω_ℓ can be written as a linear combinaison of $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial x}\frac{\partial v}{\partial x}$ and $\int_{-1}^{1}\int_{-1}^{1} \frac{\partial u}{\partial y}\frac{\partial v}{\partial y}$

so that the original problem

$$\int_\Omega k\nabla u\nabla v + \mathrm{Bi}\int_{\partial\Omega\setminus\Gamma_1} uv = \int_{\Gamma_1} v$$

can be rewritten as

$$\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$$

- where the bilinear forms a_p are parameter independent

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := v(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$A_N = \sum_{p=1}^{P} \theta_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)

all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u,v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := v(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$A_N = \sum_{p=1}^{P} \theta_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)

all the lengthy computations are thus done online and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$A_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Black-Box implementation $\sum_{p=1}^{P} g_p(\mu) a_p(u, v) = \ell(v)$

The implementation is then simplified as

- offline, the different $a_p(\zeta_i, \zeta_j)$ are precomputed,
  the $\zeta_i := u(\mu_i)$ being the reduced basis elements
- online, the computation of the stiffness matrix requires only PN^2 computations,
  instead of $\mathcal{N}^2$, where $\mathcal{N}$ is the dimension of the finite element basis.

$$\mathcal{A}_{i,j} = \sum_{p=1}^{P} g_p(\mu) a_p(\zeta_i, \zeta_j)$$

- online again the stiffness matrix is inverted in N^3 operations (direct inversion)
  all the lengthy computations are thus done offline and the ones that are done online scale with N only

Outline

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.

e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :

- then compute the following output $s = s(u)$

The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.

e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :

- then compute the following output $s = s(u)$

The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.
e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :

- then compute the following output $s = s(u)$

The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.
e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :

- then compute the following output $s = s(u)$

The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.
e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :

- then compute the following output $s = s(u)$

The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.
e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :
• then compute the following output $s = s(u)$
The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

outputs

Most of the time, the complete knowledge of the solution of partial differential equation is not required.
e.g., assume you have the following problem : find $u \in X$

$$a(u, v; \mu) =< f, v >, \quad \forall v \in X$$

What is required, generally, is outputs computed from the calculated solution :
• then compute the following output $s = s(u)$
The discretization then proceeds : find $u_\delta \in X_\delta$

$$a(u_\delta, v_\delta) =< f, v_\delta >, \quad \forall v_\delta \in X_\delta$$

and the approximated output is given by $s_\delta = s(u_\delta)$

Error Estimates

a priori convergence

How valid is the approximated output $s_\delta = s(u_\delta)$

Assuming Lipschitz condition (ex. linear case) over s, it follows that

$$|s - s_\delta| \leq c\|u - u_\delta\|_X$$

Thus any information over the error in the energy norm will allow to get verification.

Actually it is well known that the convergence of s_δ towards s most often goes faster

Error Estimates

a priori convergence

How valid is the approximated output $s_\delta = s(u_\delta)$
Assuming Lipschitz condition (ex. linear case) over s, it follows that

$$|s - s_\delta| \leq c\|u - u_\delta\|_X$$

Thus any information over the error in the energy norm will allow to get verification.
Actually it is well known that the convergence of s_δ towards s most often goes faster

Error Estimates

a priori convergence

How valid is the approximated output $s_\delta = s(u_\delta)$
Assuming Lipschitz condition (ex. linear case) over s, it follows that

$$|s - s_\delta| \leq c\|u - u_\delta\|_X$$

Thus any information over the error in the energy norm will allow to get verification.

Actually it is well known that the convergence of s_δ towards s most often goes faster

Error Estimates

a priori convergence

How valid is the approximated output $s_\delta = s(u_\delta)$
Assuming Lipschitz condition (ex. linear case) over s, it follows that

$$|s - s_\delta| \leq c\|u - u_\delta\|_X$$

Thus any information over the error in the energy norm will allow to get verification.
Actually it is well known that the convergence of s_δ towards s most often goes faster

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*

Let us assume we are in the linear output case.... $s(u) = \ell(u)$

It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \qquad (12) \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \qquad (12) \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \qquad (12) \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned} \tag{12}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \qquad (12) \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \qquad (12) \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \qquad (12) \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned} \tag{12}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*
Let us assume we are in the linear output case.... $s(u) = \ell(u)$
It is standard to introduce then the adjoint state, solution of the following problem : find $\psi \in X$

$$a(v, \psi) = -\ell(v), \quad \forall v \in X$$

Remember $a(u, \phi_\delta) = a(u_\delta, \phi_\delta) = (f, \phi_\delta)$

The error in the output is then

$$\begin{aligned} s_\delta - s &= \ell(u_\delta) - \ell(u) \\ &= a(u, \psi) - a(u_\delta, \psi) \\ &= a(u, \psi - \phi_\delta) - a(u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &= a(u - u_\delta, \psi - \phi_\delta), \quad \forall \phi_\delta \in X_\delta \\ &\leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \end{aligned} \tag{12}$$

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*

$$s_\delta - s \quad \leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \tag{13}$$

so that the best fit of ψ in X_δ can be chosen to improve the first error bound that was proposed for $|s - s_\delta|$.
For instance if ψ_δ is the solution of the Galerkin approximation to ψ, we get

$$|s - s_\delta| \leq c\|u - u_\delta\|_X \|\psi - \psi_\delta\|_X$$

But this is just *a priori* business .....

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*

$$s_\delta - s \quad \leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \tag{13}$$

so that the best fit of ψ in X_δ can be chosen to improve the first error bound that was proposed for $|s - s_\delta|$.
For instance if ψ_δ is the solution of the Galerkin approximation to ψ, we get

$$|s - s_\delta| \leq c\|u - u_\delta\|_X \|\psi - \psi_\delta\|_X$$

But this is just *a priori* business .....

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*

$$s_\delta - s \quad \leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \tag{13}$$

so that the best fit of ψ in X_δ can be chosen to improve the first error bound that was proposed for $|s - s_\delta|$.

For instance if ψ_δ is the solution of the Galerkin approximation to ψ, we get

$$|s - s_\delta| \leq c\|u - u_\delta\|_X \|\psi - \psi_\delta\|_X$$

But this is just *a priori* business .....

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*

$$s_\delta - s \quad \leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \tag{13}$$

so that the best fit of ψ in X_δ can be chosen to improve the first error bound that was proposed for $|s - s_\delta|$.
For instance if ψ_δ is the solution of the Galerkin approximation to ψ, we get

$$|s - s_\delta| \leq c\|u - u_\delta\|_X \|\psi - \psi_\delta\|_X$$

But this is just *a priori* business .....

Error Estimates

a priori convergence

The convergence of s_δ towards s most often goes faster *ctn.....*

$$s_\delta - s \quad \leq c\|u - u_\delta\|_X \|\psi - \phi_\delta\|_X, \quad \forall \phi_\delta \in X_\delta \tag{13}$$

so that the best fit of ψ in X_δ can be chosen to improve the first error bound that was proposed for $|s - s_\delta|$.
For instance if ψ_δ is the solution of the Galerkin approximation to ψ, we get

$$|s - s_\delta| \leq c\|u - u_\delta\|_X \|\psi - \psi_\delta\|_X$$

But this is just *a priori* business .....

The basic error reconstruction

In order to get *a posteriori* information, between $\ell(u_h)$ and $\ell(u_\delta)$, we have to get a hand on the residuals in the resolution (in X_δ) of the primal and dual problems. We introduce for any $v \in X$,

$$\mathcal{R}^{pr}(v;\mu) = a(u_\delta, v;\mu) - < f, v >, \quad \mathcal{R}^{du}(v;\mu) = a(v, \psi_\delta;\mu) + \ell(v).$$

Let us now compute the reconstructed errors associated to the previous residuals. These are the solutions to the following problems

$$2\alpha \int \nabla \hat{e}_h{}^{pr(du)} \nabla v_h = \mathcal{R}^{pr(du)}(v_h;\mu), \quad \forall v_h$$

we then have

Theorem Let $s^- = s_\delta - \alpha \int \nabla(\hat{e}_h{}^{pr} + \hat{e}_h{}^{pr})^2$ then $s^- \leq s_h$.

$$s_h - s^- \equiv |s_h - s_\delta|$$

How to transform to a do-able method ??
Black Box again

Outline

Selection of the parameters of the reduced basis

One possibility is to use a random approach.... and hope for the best....
.... generally it works not so badly.
But a more intelligent way can be proposed....
... based on a greedy process that combines the reduced approximation and the error evaluation.

- take a first parameter (randomly)
- use a (one dimensional) reduced basis approach over a set of parameter values (chosen randomly) and select, as a second parameter, the one for which the associated error is the largest.
  this gives now a 2 dimensional reduced basis method.
- use this (2 dimensional) reduced basis approach over the same set of parameters and select, as a third parameter, the one for which the associated error is the largest.
  this gives a 3 dimensional reduced basis method...
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Rational in front of a new problem

When you will come back home.... does your problem fits with RB approximations....

.... try step by step.

- check that your problem is parameter dependant
- compute 100 solutions with your favorite method !!!
- perform a PCA and check the eigenvalue decrease
- perform a Galerkin approximation
- try to find good parameters through projection and perform a Galerkin approximation
- and proceed...

Close the circle

- The Black Box Approach allows real time solution procedure
- Valid only for linear problem
- at least as it is.
- The use of magic points allows to tackle non linear problem
- Real life problems are non linear... with non constant coefficients

Close the circle

- The Black Box Approach allows real time solution procedure
- Valid only for linear problem
- at least as it is.
- The use of magic points allows to tackle non linear problem
- Real life problems are non linear... with non constant coefficients

Close the circle

- The Black Box Approach allows real time solution procedure
- Valid only for linear problem
- at least as it is.
- The use of magic points allows to tackle non linear problem
- Real life problems are non linear... with non constant coefficients

Close the circle

- The Black Box Approach allows real time solution procedure
- Valid only for linear problem
- at least as it is.
- The use of magic points allows to tackle non linear problem
- Real life problems are non linear... with non constant coefficients

Close the circle

- The Black Box Approach allows real time solution procedure
- Valid only for linear problem
- at least as it is.
- The use of magic points allows to tackle non linear problem
- Real life problems are non linear... with non constant coefficients

Nonlinear approximation for high-dimensional partial differential equations

$$-\Delta u(x,y) = f(x,y)$$

with homogeneous Dirichlet boundary conditions

Solution

$$g = \text{argmin}_{u \in H_0^1(\Omega)} \frac{1}{2} \int |\nabla u|^2 - \int fu$$

$$\mathcal{E}(u) = \frac{1}{2} \int |\nabla u|^2 - \int fu = \frac{1}{2} \int |\nabla (u-g)|^2 - \frac{1}{2} \int |\nabla g|^2$$

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

Extension to high dimension straightforward

Nonlinear approximation for high-dimensional partial differential equations

$$-\Delta u(x,y) = f(x,y)$$

with homogeneous Dirichlet boundary conditions
Solution

$$g = \text{argmin}_{u \in H_0^1(\Omega)} \frac{1}{2} \int |\nabla u|^2 - \int fu$$

$$\mathcal{E}(u) = \frac{1}{2} \int |\nabla u|^2 - \int fu = \frac{1}{2} \int |\nabla (u-g)|^2 - \frac{1}{2} \int |\nabla g|^2$$

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

Extension to high dimension straightforward

Nonlinear approximation for high-dimensional partial differential equations

$$-\Delta u(x,y) = f(x,y)$$

with homogeneous Dirichlet boundary conditions
Solution

$$g = \text{argmin}_{u \in H_0^1(\Omega)} \frac{1}{2} \int |\nabla u|^2 - \int fu$$

$$\mathcal{E}(u) = \frac{1}{2} \int |\nabla u|^2 - \int fu = \frac{1}{2} \int |\nabla (u-g)|^2 - \frac{1}{2} \int |\nabla g|^2$$

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

Extension to high dimension straightforward

Nonlinear approximation for high-dimensional partial differential equations

$$-\Delta u(x,y) = f(x,y)$$

with homogeneous Dirichlet boundary conditions
Solution

$$g = \text{argmin}_{u \in H_0^1(\Omega)} \frac{1}{2} \int |\nabla u|^2 - \int fu$$

$$\mathcal{E}(u) = \frac{1}{2} \int |\nabla u|^2 - \int fu = \frac{1}{2} \int |\nabla (u-g)|^2 - \frac{1}{2} \int |\nabla g|^2$$

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

Extension to high dimension straightforward

Nonlinear approximation for high-dimensional partial differential equations

$$-\Delta u(x,y) = f(x,y)$$

with homogeneous Dirichlet boundary conditions
Solution

$$g = \text{argmin}_{u \in H_0^1(\Omega)} \frac{1}{2} \int |\nabla u|^2 - \int fu$$

$$\mathcal{E}(u) = \frac{1}{2} \int |\nabla u|^2 - \int fu = \frac{1}{2} \int |\nabla (u-g)|^2 - \frac{1}{2} \int |\nabla g|^2$$

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

Extension to high dimension straightforward

Nonlinear approximation for high-dimensional partial differential equations

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

pure greedy
orthogonal greedy

Nonlinear approximation for high-dimensional partial differential equations

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

pure greedy

orthogonal greedy

Nonlinear approximation for high-dimensional partial differential equations

$$u = \sum_{n=1}^{\infty} r_n(x) s_n(y)$$

pure greedy
orthogonal greedy

That's all folks

Thanks !!

That's all folks

Thanks !!