Prof. Dr. Karsten Urban
Kristina Steih

SS 2012
30.05.2012

NumPDE II - Blatt 6

Nachdem wir jetzt eine reduzierte Basis mittels des Greedy-Algorithmus erstellen können und auch einen effizienten Fehlerschätzer für die Zustandsvariable u haben, sollen nun Fehlerschätzer für das lineare Output-Funktional

$$s(\mu) = \ell(u(\mu)) := \int_{\bar{\Omega}} u(x;\mu)dx, \qquad \bar{\Omega} := [0.4, 0.6] \times [0.4, 0.6],$$

implementiert werden.

Vorbereitung:

Im Material zu diesem Blatt sind die folgenden Funktionen leicht verändert worden, um die FE-Größen für das Output-Funktional zu erhalten: `getInitFEM`, `performPreAssembling` und `assembleFemPre`. Außerdem gibt es noch die zusätzliche Funktion `getDualSol`, sowie eine Hilfsfunktion `initFEMOutput`.

- Schauen Sie sich kurz die Änderungen an, insbesondere die Speicherung der benötigten vorassemblierten Größen in der Funktion `performPreAssembling`.

Aufgabe 1 *(Output-Fehlerschätzer, simple Version)*

Der einfachste Fehlerschätzer für $s_N(\mu) := \ell(u_N(\mu))$ ist gegeben durch

$$s^{\mathcal{N}}(\mu) - s_N(\mu) \;\leq\; \Delta_N^{\text{out}}(\mu) := \|\ell\|_{X'} \cdot \Delta_N(\mu).$$

a) Wie kann für gegebene RB-Lösungskoeffizienten $\underline{u}_N(\mu)$ die Output-Approximation $\ell(u_N(\mu))$ in $\mathcal{O}(N)$ Operationen berechnet werden? Wie erhält man $\|\ell\|_{X'}$ und damit den Fehlerschätzer?

Hinweis: siehe Behandlung der rechten Seite f der PDE.

b) Ergänzen Sie die Berechnung der nötigen Offline-Komponenten im Greedy-Training.

c) Ergänzen Sie die Berechnung der Approximation $s_N(\mu)$ in Zeile 93 im Skript `blatt6.m`.

d) Schreiben Sie die (einzeilige) Funktion `getErrEstOut_simple`, welche $\Delta_N^{\text{out}}(\mu)$ berechnen soll, und vergleichen Sie Fehler und Fehlerschätzer (Zeilen 96-97 in `blatt6.m`).

Aufgabe 2 *(Output-Fehlerschätzer, primal-duale Version)*

Der Fehlerschätzer aus Aufgabe 1 ist oft ziemlich grob. Eine Alternative bieten primal-duale Ansätze. Dabei betrachten wir die Output-Approximation

$$s_{N,\widetilde{N}}(\mu) := \ell(u_N(\mu)) - r_N(z_{\widetilde{N}}(\mu);\mu),$$

wobei $r_N(\cdot;\mu)$ das primale Residuum und $z_{\widetilde{N}}$ eine Lösung des reduzierten dualen Problems

$$a(w, z_{\widetilde{N}}(\mu);\mu) = -\ell(w) \qquad \forall\, w \in \widetilde{X}_{\widetilde{N}}$$

ist. (Notation dualer Fehlerschätzer: $\|z(\mu) - z_N(\mu)\| \leq \widetilde{\Delta}_{\widetilde{N}}(\mu)$.)

a) Zeigen Sie, dass sich der primal-duale Fehlerschätzer aus Proposition 9.12 noch verbessern lässt zu

$$s^{\mathcal{N}}(\mu) - s_{N,\widetilde{N}}(\mu) \;\leq\; \Delta^{\text{out}}_{N,\widetilde{N}}(\mu) := \alpha_{\text{LB}}(\mu)\Delta_N(\mu)\widetilde{\Delta}_{\widetilde{N}}(\mu).$$

Hinweis: Es gilt $r_N(z_{\widetilde{N}};\mu) = a(e_N, z_{\widetilde{N}};\mu)$ (warum?).

b) Es gibt verschiedene Vorgehensweisen zur Konstruktion von $\widetilde{X}_{\widetilde{N}}$. Wir werden hier den dualen RB-Raum völlig unabhängig vom primalen Raum X_N erzeugen.
Schreiben Sie eine Funktion `trainDualRB_Greedy`, welche das Greedy-Training für $\widetilde{X}_{\widetilde{N}}$ durchführt.

Hinweis: Die Funktion `trainRB_Greedy` *kann dazu zum allergrößten Teil einfach kopiert werden.*

c) Wie kann $s_{N,\widetilde{N}}(\mu)$ online berechnet werden (in Matrix-Vektor-Schreibweise)? Welche Größen müssen dafür offline noch aufgestellt werden? Ergänzen Sie diese im dualen Greedy und schreiben Sie eine Funktion `getOutputAppr` zur Berechnung von $s_{N,\widetilde{N}}(\mu)$.

d) Implementieren Sie nun noch den Einzeiler `getErrEstOut_primaldual` zur Berechnung von $\Delta^{\text{out}}_{N,\widetilde{N}}(\mu)$ und vergleichen Sie wieder Fehler und Fehlerschätzer sowie die Effektivitäten beider Ansätze (Zeilen 146-166 in `blatt6.m`).