Model Order Reduction
Problem Set — Parabolic Problems

Christophe Prud'homme

December 19, 2019

Our problem of interest is the thermal fin discussed in the previous problem sets, but now we consider the time-dependent case. We assume that the thermal fin is initially at zero (non-dimensionalized) temperature and a heat flux is then applied to the root. The output of interest is the average temperature of the fin. We directly consider the truth approximation. To this end, we divide the time interval, $I = (0, t_f]$, into K subintervals of equal length $\Delta = \frac{t_f}{K}$, and define $t_k = k\Delta t, 0 \leq k \leq K$. We shall consider Euler-Backward for the time integration. We also recall the truth finite element approximation space $X \subset X^e$.

Our truth problem statement is then: given a parameter $\mu \in D$, we evaluate the output

$$s^k(\mu) = l(u^k(\mu)), 1 \leq k \leq K, \tag{1}$$

where the field variable $u^k(\mu) \in X, 1 \leq k \leq K$, satisfies

$$m\left(\frac{u^k(\mu) - u^{k-1}(\mu)}{\Delta t}, v\right) + a(u^k(\mu), v; \mu) = f(v)g(t^k), \forall v \in X \tag{2}$$

with initial condition $u(t_0; \mu) = u_0 = 0$. Here, the bilinear form a is defined as in Problem Set 1, the linear form f is given by $f(v) = \int_{\Gamma_{root}} v$, the linear form l is given by $l(v) = \int_\Omega v$, the bilinear form m is given by

$$m(u, v) = \int_\Omega uv, \forall u, v \in X \tag{3}$$

and $g(t_k)$ denotes the "control input" at time $t = t_k$. Note that m and l, f are parameter-independent.

We consider the following special case: We assume that the conductivities of all fins are equivalent and fixed at $k_i = 1, i = 1, ..., 4$, and that the Biot number is allowed to vary between 0.01 and 1. We thus have $\mu \equiv Bi \in D = [0.01, 1]$. We consider the time interval $I = (0, 10]$ with a discrete timestep $\Delta t = 0.1$ and thus $K = 100$.

To begin, you should download and unpack the zip file `PS4_matlab.zip`. You will find the file `FE_matrix_mass.mat` which contains a struct, `FE_matrix_mass`, with the mass matrices for the fine, medium, and coarse triangulations used before. To generate the output vector L you can simply postmultiply the corresponding mass matrix with a vector containing all 1s. From the previous problem sets you already have the required finite element forcing vector F and the finite element stiffness matrix A (and the A_q). In the sequel, you should use the medium triangulation.

1 Part 1 - Reduced Basis Approximation

We first generate a reduced basis approximation by choosing a basis from scratch. To this end, we use $g(t_k) = \delta_{1k}, 1 \le k \le 100$ (unit impulse input) and set

$$X_N = span\left\{u^1(0.01), u^5(0.01), u^{10}(0.01), u^{20}(0.01), u^{30}(0.01), u^5(0.1), u10(0.1), u^{20}(0.1), u^5(1), u^{10}(1)\right\}, \tag{4}$$

i.e., our reduced basis space X_N is spanned by the solution $u^k(\mu)$ at several parameter-time pairs. We then orthonormalize X_N using Gram-Schmidt.

Q1. Write an offline-online code in matlab for the reduced basis approximation (use LU decomposition for the truth and reduced basis time integration).

- Plot the outputs $s^k(\mu)$, $s_N^k(\mu)$, and the error $s^k(\mu) - s_N^k(\mu)$ as a function of time for $g(t_k) = 1 - cos(t_k)$ and $\mu = 0.05$.
- Plot $|||u^k(\mu)|||$, $|||u_N^k(\mu)|||$, and the error $|||u^k(\mu) - u_N^k(\mu)|||$ as a function of time for $g(t_k) = 1 - cos(t_k)$ and $\mu = 0.05$.

2 Part 2 - A Posterior Error Estimation

The problem statement fits in the framework introduced in the lecture. Q2. Similar to the elliptic case, we can compute the energy norm bound directaly from the residual (N-dependent cost) or we can use the offline-online decomposition.

- Derive and implement an offline-online version for the calculation of the energy norm a posteriori error bound for the primal variable by extending your code from the elliptic case. **Note:** we will concentrate on the energy norm bound for the primal variable here, so you do not need to consider the dual problem (reduced basis approximation or a posteriori error estimation). Also, we will use the simple output bound and we thus do not require the residual correction term.
- Compare the direct calculation of the error bound with your offline-online decomposition for 10 random parameter values in D. You can perform a comparison over time (better) or compare the values at the final time.

3 Part 3 - Sampling Procedure

Our reduced basis space from Part 1 is less than optimal. Given your offline-online decomposition for the reduced basis approximation from Part 1 and associated a posteriori error estimation from Part 2 we can now pick a much more optimal basis. Q3. Apply the POD-Greedy algorithm with $\Xi_{train} = G^{ln}_{[0.01,1;100]}$, $\varepsilon^{tol,min} = 1e-6$, and $\mu_0^* = 0.01$. Here, we also use the impulse input $g(t_k) = \delta_{1k}, 1 \le k \le 100$.

- What is the value of N_{max} to achieve the desired accuracy?
- Plot $\Delta_N^{max} = \Delta_N^K(\mu^*)/|||u^k(\mu^*)|||$ as a function of N
- Plot the outputs $s^k(\mu)$, $s_N^k(\mu)$, the error $s^k(\mu) - s_N^k(\mu)$, and the simple error bound $\Delta_N^s(t_k;\mu)$ as a function of time for $N = 10$ and $N = Nmax$ for $g(t_k) = 1 - cos(t_k)$ and $\mu = 0.05$.

- Plot $|||u^k(\mu)|||$, $|||u_N^k(\mu)|||$, the error $|||u^k(\mu) - u_N^k(\mu)|||$, and the error bound $\Delta_N(t_k;\mu)$ as a function of time for $N = 10$ and $N = Nmax$ for $g(t_k) = 1 - cos(t_k)$ and $\mu = 0.05$.

- Calculate the average effectivity $\bar{\eta}^\mu$ for $\Xi_{test} = G^{ln}_{[0.01,1;15]}$ and $g(t_k) = 1 - cos(t_k)$ as a function of N.

- Compare the average online time to calculate $s_N(t_k;\mu)$ and $\Delta_N^s(t_k;\mu)$ for $\Xi_test = G^{ln}_{[0.01,1;15]}$ with the time for direct calculation of $s(t_k;\mu)$ (Choose N based on (c) such that the error in the output bound is approximately 1%).