Scientific Computing for PDEs

Homework set 1:

Finite Elements and Model Order Reduction in dimension one

Introduction

This project guides you to the concept of numerical methods for parametrized PDEs, which is a field of research in itself, usually known under the name of Model Order Reduction. To motivate it, and illustrate several applications, we consider the following problem in $\Omega = (0, 1)$:

$$\begin{aligned} &\text{Find } u \in V = H^1_0(\Omega) \text{ such that} \\ &\nu \int_\Omega u'v' \,\mathrm{d}x + \gamma \int_\Omega uv \,\mathrm{d}x + \beta \int_\Omega u'v \,\mathrm{d}x = \int_\Omega fv \,\mathrm{d}x, \quad \forall v \in V, \end{aligned} \tag{1}$$

where $f \in L^2(\Omega)$ and ν, β and γ are real-valued parameters. We gather them in a vector $\mu = (\nu, \gamma, \beta)$, and we assume that they can take values in a compact set $\mathcal{P} \subset \mathbb{R}^3$. When the vector μ of parameters varies, the solution u_μ also varies, and we record this dependence through the subscript μ. In applications, it is often necessary to solve the PDE for many different parameter values $\mu \in \mathcal{P}$. Since FEM solvers can become very computationally expensive when the problem is very complex, it becomes important to speed up the mapping between parameters and solutions. This is typically important in the following cases:

1. **Uncertainty propagation:** When the parameters are uncertain (due to, for example, an imperfect characterization from experimental data), one may need to describe μ as a probability distribution. This induces a distribution in the PDE solutions u_μ that one would like to characterize (in terms of, e.g., expectation or variance). To do this numerically, we need to sample the solution u_μ for many values of μ.
2. **Parameter optimization:** In design applications, one may need to find the value of the parameter μ^* that best optimizes a certain criterion. Such problems come in the form of

   $$\min_{\mu \in \mathcal{P}} \mathcal{L}(u_\mu)$$

   where $\mathcal{L} : V \longrightarrow \mathbb{R}_+$ is a loss function that is designed following the needs of each application.
3. **Inverse problems:** In other situations, we may know the PDE but not the exact parameter μ that represents the physical process. Instead, we may have access to partial observation of the solution, for example, evaluations $u(x_i)$ at certain points x_i for $1 \leq i \leq m$. These observations may be given by sensors placed in the domain. In this context, one may want to reconstruct u from the observations, or find the parameter μ that best explains what is observed. In both cases, Reduced Order Modeling plays an important role in developing numerical algorithms.

The main goal in Reduced Order Modelling is to approximate the set of parametric solutions

$$\mathcal{M} := \{u_\mu \;:\; \mu \in \mathcal{P}\} \subset V$$

with a linear space $V_n \subset V$ of small dimension n. The quality of a given V_n to approximate $\mathcal{M}$ can be evaluated via the quantity

$$e(\mathcal{M}, V_n) = \max_{\mu \in \mathcal{P}} \|u_\mu - u_{n,\mu}\|_V,$$

where $u_{n,\mu}$ is the solution obtained with the Galerkin method that uses V_n for the test and the trial space. We wish to build a sequence of subspaces $(V_n^{\text{rom}})_{n\geq 1}$ for which $e(\mathcal{M}, V_n^{\text{rom}})$ decays fast as $n \to \infty$. Ideally, this sequence of subspaces should give a much better accuracy than the obtained with FEM spaces V_n^{fem} with n degrees of freedom in the sense that:

$$e(\mathcal{M}, V_n^{\text{rom}}) \ll e(\mathcal{M}, V_n^{\text{fem}}) \quad \forall n \in \mathbb{N}.$$

In the following, we work with the $H_0^1(\Omega)$-norm,

$$\|v\|_V := \left(\|v\|^2_{L^2(\Omega)} + \|v'\|^2_{L^2(\Omega)}\right)^{1/2}.$$

1 Building a FEM solver

In this first part, we focus on analyzing the variational formulation (1), and on building a FEM solver for it.

1. Show that problem (1) has a unique solution under certain conditions on the parameters μ that you will specify.
2. Derive a strong formulation for problem (1) and specify the regularity assumptions that are needed for u and f.
3. In the following, we fix $f = 1$, and we consider an approximation of the solution with $\mathbb{P}_1(\Omega)$ finite elements of degree 1. The interval Ω is meshed with a uniform grid involving N cells. We thus have $N + 1$ grid points

   $$x_i = ih, \ 0 \leq i \leq N$$

   with $h = N^{-1}$. The associated finite element space can be expressed as

   $$V_N^{\text{fem}} = \text{span}\{\varphi_i\}_{i=1}^{N-1} \subset H_0^1(\Omega)$$

   where the φ_i are the hat functions that we defined in the lecture notes.

   If we express the finite element solution $u_{N,\mu}^{\text{fem}}$ as

   $$u_{N,\mu}^{\text{fem}}(x) = \sum_{i=1}^{N-1} c_{i,\mu}^{\text{fem}} \varphi_i(x),$$

   show that the vector $c_\mu^{\text{fem}} = (c_{i,\mu}^{\text{fem}})_{i=1}^{N-1}$ satisfies the linear system

   $$A_\mu^{\text{fem}} c_\mu^{\text{fem}} = b^{\text{fem}}$$

   for a matrix $A_\mu^{\text{fem}} \in \mathbb{R}^{(N-1)\times(N-1)}$ and a vector $b^{\text{fem}} \in \mathbb{R}^{N-1}$ that you will specify. In particular, show that

   $$A_\mu^{\text{fem}} = \nu B_1^{\text{fem}} + \gamma B_2^{\text{fem}} + \beta B_3^{\text{fem}}$$

   where $B_1^{\text{fem}}, B_2^{\text{fem}}, B_3^{\text{fem}}$ are $(N-1) \times (N-1)$ matrices independent of μ.

4. Write a Python code that computes, for a given parameter μ and a given number of degrees of freedom N, the solution $u^{\mathrm{fem}}_{N,\mu}$.

5. To validate the code, we consider the case where $\gamma = 0$ and $f = 1$.

   (a) Solve the strong formulation of the problem, and show that it has an explicit expression. Show that this is the unique solution to the weak formulation.

   (b) Visualization:

      i. Plot the exact solution u and the FEM solution $u^{\mathrm{fem}}_{N,\mu}$ for $\beta = 1$ and $\nu = 1$, and for $N \in \{10^1, 10^2, 10^3, 10^4\}$. Same question for $\nu = 10^{-3}$ and $\nu = 10^{-4}$.

      ii. What do you observe around $x = 1$? Can you connect this behavior with the a priori convergence result from Theorem 3.14 of the lecture notes?

      iii. Does the choice of a uniform mesh seem pertinent to you? What would be an alternative?

      iv. What happens to the solution depending on the sign of β? What is the physical interpretation?

   (c) Convergence study in h. Derive a strategy to estimate numerically the errors

$$e^{L^2(\Omega)}_{\mu}(N) = \|u - u^{\mathrm{fem}}_{N,\mu}\|_{L^2(\Omega)},$$
$$e^{H^1_0(\Omega)}_{\mu}(N) = \left(\|u - u^{\mathrm{fem}}_{N,\mu}\|^2_{L^2(\Omega)} + \|u' - (u^{\mathrm{fem}}_{N,\mu})'\|^2_{L^2(\Omega)}\right)^{1/2}.$$

   Make a plot of the above quantities as N varies in $[10^1, 10^4]$, and comment on the observed behavior. Does your observations match the convergence result from Theorem 3.9?

2 Reduced Order Modeling

We now turn to the case in which the parameters $\mu = (\nu, \gamma, \beta)$ vary. We consider

$$\mathcal{P} = [10^{-1}, 1] \times [0, 1] \times [0, 1], \quad \mathcal{M} = \{u_\mu \ : \ \mu \in \mathcal{P}\}.$$

As already explained in the introduction, the goal is to build a sequence of spaces $(V_n)_{n\in\mathbb{N}}$ such that $e(\mathcal{M}, V_n)$ decays fast as $n \to \infty$. We consider constructions in which

$$V^{\mathrm{rom}}_n = \mathrm{span}\{u_i\}_{i=1}^n,$$

where $u_i = u^{\mathrm{fem}}_{N,\mu_i}$ for some $\mu_i \in \mathcal{P}$. Of course, the choice of the specific μ_i will have a big impact on the decay rate of $e(\mathcal{M}, V^{\mathrm{rom}}_n)$. The spatial discretization is assumed to be fine enough so N is assumed to be a rather large number (we will give indications on how to choose it later on).

2.1 Questions

1. **Theoretical preparation:**

   (a) Write the Galerkin variational formulation of problem (1) when we use V^{rom}_n as the trial and the test space, and show that it is well posed for all $\mu \in \mathcal{P}$. We denote by $u^{\mathrm{rom}}_{n,\mu}$ the solution for each μ.

   (b) Give an upper bound of the V-error between u_μ and $u^{\mathrm{rom}}_{n,\mu}$.

(c) Show that the variational formulation is equivalent to solving a linear system

$$A_\mu^{\mathrm{rom}} c_\mu^{\mathrm{rom}} = b^{\mathrm{rom}} \tag{2}$$

of size $n \times n$, where you will explicitly give the form of A_μ^{rom} and b^{rom}. In particular, show that

$$A_\mu^{\mathrm{rom}} = \nu B_1^{\mathrm{rom}} + \gamma B_2^{\mathrm{rom}} + \beta B_3^{\mathrm{rom}}$$

where B_1^{rom}, B_2^{rom}, B_3^{rom} are $n \times n$ matrices independent of μ.

(d) In Reduced Order Modeling, B_1^{rom}, B_2^{rom}, B_3^{rom} and b^{rom} are computed and saved once and for all. Explain how to compute them in practice using the vectors $c_{\mu_i}^{\mathrm{fem}}$, and the matrices B_1^{fem}, B_2^{fem}, B_3^{fem} from question 3 of Section 1.

(e) Complexity analysis: Compare the numerical complexity to solve K problems for K different values of μ with the direct FEM method, and with the reduced basis method. The analysis should be done in terms of the number of degrees of freedom involved, and the number of operations to solve the linear problems.

2. **Implement the Reduced Order Modeling method.** Here are some hints to structure your code:

   (a) Fix $N \gg 1$.

   (b) Assemble B_1^{fem}, B_2^{fem}, B_3^{fem} and b^{fem} from question 3 of Section 1.

   (c) Compute $V_n^{\mathrm{rom}} = \mathrm{span}\{u_i\}_{i=1}^n$ with $u_i = u_{N,\mu_i}^{\mathrm{fem}}$, and for the μ_i obtained in the procedures that we explain later on in Section 2.2. Store the solution coefficients $c_{\mu_i}^{\mathrm{fem}} \in \mathbb{R}^N$ in a big matrix $(c_{\mu_1}^{\mathrm{fem}} | \dots | c_{\mu_n}^{\mathrm{fem}}) \in \mathbb{R}^{N \times n}$.

   (d) Assemble and store the matrices B_1^{rom}, B_2^{rom}, B_3^{rom} and b^{rom}.

   (e) For every $\mu \in \mathcal{P}$, compute $u_{n,\mu}^{\mathrm{rom}}$ by solving the linear system (2).

3. **Convergence study:** We wish to study the convergence of $e(\mathcal{M}, V_n^{\mathrm{rom}})$ as n increases. The goal of this question is to make a plot of this quantity when $n \in \{2, \dots, 50\}$. Here, V_n^{rom} means the space obtained with the naive approach and the greedy approach from Section 2.2. We take $N = 10^3$ degrees of freedom for the FEM solver involved in the construction of V_n^{rom}.

   To estimate $e(\mathcal{M}, V_n^{\mathrm{rom}})$, make a random draw of a subset $\widetilde{\mathcal{P}}$ of 500 samples and approximate

   $$e(\mathcal{M}, V_n^{\mathrm{rom}}) = \max_{\mu \in \mathcal{P}} \|u_\mu - u_{n,\mu}^{\mathrm{rom}}\|_V \approx \max_{\mu \in \widetilde{\mathcal{P}}} \|u_{N,\mu}^{\mathrm{fem}} - u_{n,\mu}^{\mathrm{rom}}\|_V.$$

4. How does the behavior of the convergence rate change when we consider

   $$\widehat{\mathcal{P}} = [10^{-2}, 1] \times [0,1] \times [0,1]$$

   as the parameter set?

2.2 Construction of V_n^{rom}

We propose to explore two ways of building $V_n^{\mathrm{rom}} = \mathrm{span}\{u_{N,\mu_i}^{\mathrm{fem}}\}_{i=1}^n$:

1. **Naive approach:** We pick randomly n parameters $\{\mu_1, \dots, \mu_n\}$ with $\mu_i \in \mathcal{P}$. We compute the associated FEM solutions $u_i = u_{N,\mu_i}^{\mathrm{fem}}$ and define $V_n^{\mathrm{rom}} = \mathrm{span}\{u_i\}_{i=1}^n$.

2. **Greedy algorithm:** We build a training set $\widetilde{\mathcal{P}} = \{\mu_1, \dots, \mu_{500}\}$ containing 500 parameter values drawn randomly from a uniform distribution in $\mathcal{P}$. Fix N and compute the subset

   $$\widetilde{\mathcal{M}} = \{u^{\text{fem}}_{N,\mu} \ : \ \mu \in \widetilde{\mathcal{P}}\}.$$

   We next proceed iteratively as follows:

   - $n = 1$: Select
     $$\mu_1 \in \arg\max_{\mu \in \widetilde{\mathcal{P}}} \|u^{\text{fem}}_{N,\mu}\|_V,$$
     and set $V^{\text{rom}}_1 = \text{span}\{u^{\text{fem}}_{N,\mu_1}\}$.
   - $n > 1$: Assume that we have built V^{rom}_{n-1}. Compute
     $$\mu_n \in \arg\max_{\mu \in \widetilde{\mathcal{M}}} \|u^{\text{fem}}_{N,\mu} - u^{\text{rom}}_{n-1,\mu}\|_V,$$
     and set $V^{\text{rom}}_n = \text{span}\{u^{\text{fem}}_{N,\mu_i}\}_{i=1}^n$.

3 Evaluation of the code

Below are some elements that we will consider to evaluate the quality of your code beyond having it running and giving correct answers:

1. the code is concise but the commands are easily readable,
2. well commented and documented,
3. efficient use of libraries such as numpy and scipy,
4. efficient use of classes and functions to:
   - avoid code duplication
   - enhance readability,
   - make the code reproduce as close as possible the main conceptual building blocks.