Numerical Optimization of Partial Differential Equations

Part I: generalis

Bartosz Protas

Department of Mathematics & Statistics
McMaster University, Hamilton, Ontario, Canada
URL: http://www.math.mcmaster.ca/bprotas

The Isaac Newton Institute
January 25–26, 2022

Acknowledgments

- Pritpal 'Pip' Matharu
  (Ph.D. student)

Applications of PDE Optimization

- Open-loop optimal control of distributed systems
  - flow control problems in fluid mechanics (e.g., optimization of lift and/or drag, mixing, etc.)
  - structural optimization is solid mechanics
  - process optimization in chemical engineering
  - portfolio optimization in investing
- State and parameter estimation for distributed systems
  - inverse problems for PDEs (e.g., medical imaging)
  - data assimilation in Numerical Weather Prediction ("4D VAR")

General Framework

- Equation-constrained optimization problem

$$(\star)\qquad \begin{cases} \inf_{(x,\varphi)} \widetilde{\mathcal{J}}(x,\varphi) \\ \text{subject to:} \quad S(x,\varphi) = 0 \end{cases}$$

where:

- $x \in \mathcal{X}$ — the state variable ($\mathcal{X}$ is a suitable function space)
- $\varphi \in \mathcal{U}$ — the control variable ($\mathcal{U}$ is a suitable function (Hilbert) space)
- $\widetilde{\mathcal{J}} \; : \; \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$ — the objective functional
- $S \; : \; \mathcal{X} \times \mathcal{U} \rightarrow \mathcal{X}^*$ — constraint (PDE with initial/boundary conditions)

- The constraint $S(x,\varphi) = 0$ be handled by introducing the *Lagrange multiplier* $\lambda \in \mathcal{X}$, such that we can define the Lagrangian

$$\mathcal{L}(x,\varphi,\lambda) = \widetilde{\mathcal{J}}(x,\varphi) - \langle \lambda, S(x,\varphi) \rangle_{\mathcal{X}\times\mathcal{X}^*}$$

- The constrained minimizers are then defined by the variational problem

$$\sup_{\lambda\in\mathcal{X}} \inf_{(x,\varphi)\in\mathcal{X}\times\mathcal{U}} \mathcal{L}(x,\varphi,\lambda)$$

- Stationary points $(\widetilde{x},\widetilde{\varphi},\widetilde{\lambda})$ of the Lagrangian are solutions of the Euler-Lagrange equations

$$\nabla_\lambda \mathcal{L}(\widetilde{x},\widetilde{\varphi},\widetilde{\lambda}) = 0$$
$$\nabla_x \mathcal{L}(\widetilde{x},\widetilde{\varphi},\widetilde{\lambda}) = 0$$
$$\nabla_\varphi \mathcal{L}(\widetilde{x},\widetilde{\varphi},\widetilde{\lambda}) = 0$$

- The stationary points $(\widetilde{x},\widetilde{\varphi},\widetilde{\lambda})$ are saddle points. The problem is hard so solve and we will advocate for a different formulation.

- If the constraint equation $S(x, \varphi) = 0$ can be solved for x (cf. implicit function theorem), then $x = x(\varphi)$ and one can define the *reduced* objective functional

$$\mathcal{J}(\varphi) := \widetilde{\mathcal{J}}(x(\varphi), \varphi)$$

- Constrained optimization problem $(\star)$ can then replaced with the following equivalent unconstrained problem

$$\min_{\varphi \in \mathcal{U}} \mathcal{J}(\varphi)$$

- *Inequality* constraints are more difficult to handle, especially in the context of PDE optimization, and will not be considered here

- How to find a local minimizer $\widetilde{\varphi}$?

- Consider the following initial-value problem in the space $\mathcal{U}$, known as the *gradient flow*

$$\text{(GF)} \qquad \begin{cases} \dfrac{d\varphi(\tau)}{d\tau} = -\nabla\widetilde{\mathcal{J}}(\varphi(\tau)), & \tau > 0, \\ \varphi(0) = \varphi_0, \end{cases}$$

  where
  - τ is a "pseudo-time" (a parametrization)
  - φ_0 is a suitable initial guess

- Then, $\lim_{\tau\to\infty} \varphi(\tau) = \widetilde{\varphi}$

- When the optimization is nonconvex, “solution” mean a *local minimizer*
  - one is often interested in branches of local maximizers obtained as some parameter is varied

- In principle, the gradient flow may converge to a *saddle point* φ_s, where $\nabla\widetilde{\mathcal{J}}(\varphi_s) = 0$ and the Hessian $\nabla^2\widetilde{\mathcal{J}}(\varphi_s)$ is *not* positive-definite, but in actual computations this is very unlikely.

- Optimize-then-Discretize: optimality conditions and gradient expressions derived at the continuous (PDE) level and only then discretized $\Longleftarrow$ will focus on this approach
  - formulation independent of discretization
  - allows one to exploit the analytic structure of the problem (e.g., regularity, etc.)
  - works well with mesh refinement in the numerical solution of PDEs
- Discretize-then-Optimize: the PDE problem is discretized first and then treated as optimization problem in finite dimension
  - PDE discretization errors do not affect the optimization procedure
  - can take advantage of Automatic Differentiation (AD) tools
  - may be more suitable for very large problems