Inverse Problems

Sonia Reilly

October 20, 2023

These notes are based on Georg Stadler's Fall 2021 lecture notes, as well as Noemi Petra and Georg Stadler's "Model Variational Inverse Problems Governed by PDEs".

1 Inverse problems and PDE-constrained optimization

- We typically re-formulate an inverse problem of the form

$$F(m) + e = d,$$

  where m is the parameter we solve for and e is error, into an optimization problem of the form

$$\begin{aligned}
&\min_{m,u} J(u, m) \\
&c(u, m) = 0, \\
&+ \text{ constraints, e.g., } m \leq b.
\end{aligned}$$

  Here the $c(u, m) = 0$ is a PDE with solution u and parameters m.

- To solve for m using regularization, we solve the (PDE-constrained) optimization problem

$$\min_m \frac{1}{2}||F(m) - d||^2 + R(m).$$

- Alternative: Bayesian framework where m, e are random variables and we solve for posterior distribution.

2 Ill-posedness and regularization

- Well-posed means there exists a unique solution that depends continuously on the data.
- Inverse problems are ill-posed (example: in image blurring, a high frequency perturbation in the image leads to almost no perturbation in the blurred image, so the inverse problem is unstable.)
- Inverse heat equation is ill-posed because of the stiffness of the operator – smallest eigenvalues are tiny, so if we invert the system the corresponding high frequency modes of the noise will be hugely amplified.
- Regularization in the eigendecomposition of F^{-1}

- If λ_i, v_i are the eigenvalues and eigenvectors of the forward operator F, then $d = \sum_i \alpha_i v_i$, so $m \approx \sum_i \frac{\alpha_i}{\lambda_i} v_i$. We add a weight $w(\lambda_i^2)$ to dampen the high frequency modes.
- Truncated SVD: $w_\alpha(\lambda^2)$ is 0 for $\lambda_i \leq \alpha$, otherwise 1. For a symmetric pos def operator, the SVD is the same as the eigenvalue decomposition.
- Tikhonov regularization: $w_\alpha(\lambda^2) = \frac{\lambda^2}{\lambda^2+\alpha^2}$ – smooth approximation to the abrupt filtering of TSVD.

• Regularization in the objective function

  - We can also solve the ill-posedness by adding a regularization term to the objective function.
  - Tikhonov regularization: The minimization problem

$$\min_m \frac{1}{2}||F(m) - d||^2 + \frac{\alpha}{2}||m||^2$$

    is equivalent to the Tikhonov regularization in its singular value form. (Note that the solution to the normal equations for this minimization is

$$m_{TK} = (F^T F + \alpha I)^{-1} F^T d,$$

    so we can see the effect of α to make $F^T F$ better conditioned.)
  - L^1-regularization (compressive sensing): continuous $\alpha \int_\Omega |m(x)|dx$, discretized $\alpha||m||_{\ell_1}$. Encourages sparsity in m.
  - Tikhonov-H^1/quadratic gradient: continuous $\frac{\alpha}{2}\int_\Omega |\nabla m|^2 dx$, discrete $\frac{\alpha}{2}\sum(m_{i+1} - m_i)^2$. Encourages smoothness in m.
  - Total variation (TV): continuous $\alpha \int_\Omega |\nabla m| dx$, discrete $\alpha \sum(m_{i+1} - m_i)$, encourages sparsity in ∇m, so useful when solution is expected to be piecewise smooth.

• Choosing the regularization parameter α

  - All methods require solving the regularized inverse problems many times for different α
  - L-curve criterion: Plot $||Fm_\alpha - d||$ vs. $||m_\alpha||$ on a log-log plot and choose the α at the "bend" of the resulting L-shaped curve (point of highest curvature). Finds a tradeoff point between low error and low $||m||$. Good in practice but can't prove that $\alpha^* \to 0$ as $||e|| \to 0$.
  - Morozov's discrepancy principle: If noise level $||e||$ is known, choose the maximum α such that $||Fm_\alpha - d|| \leq ||e||$, to prevent overfitting. Can prove that $\alpha^* \to 0$ as $||e|| \to 0$.

3 Examples of PDE Inverse Problems

3.1 Inverse Advection-Diffusion

• Forward problem:

$$\begin{cases} u_t + v \cdot \nabla u - \kappa \Delta u = 0 & \text{on } \Omega \times [0,T] \\ u(x,0) = u_0(x) & \text{on } \Omega \times t = 0 \\ \kappa \nabla u \cdot \hat{n} = 0 & \text{on } \partial\Omega \times [0,T]. \end{cases}$$

- Inverse problem: Given noisy measurements $u_d(x)$ along the boundary at time T, solve for the initial condition $u_0(x)$. With Tikhonov regularization, this is the minimization problem

$$\min_{u_0(x)} \frac{1}{2}\int_{\partial\Omega}(u(x,T)-u_d(x))^2dx + \frac{\alpha}{2}\int_{\Omega}(u_0(x))^2dx$$

subject to the forward PDE as a constraint.

3.2 Elliptic parameter estimation

- Forward problem:

$$\begin{cases} -\nabla(a(x)\nabla u) = f & \text{on } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases}$$

- Possible interpretations: Steady state heat equation (temperature u, thermal conductivity a, heat source f), groundwater filtration (pressure u, permeability a, gravity force f), membrane with varying stiffness (vertical displacement u, stiffness a, force f).

- With Tikhonov-H^1 regularization, the inverse problem is

$$\min_{a(x)} \frac{1}{2}\int_{\Omega}(u-u_d)^2dx + \frac{\alpha}{2}\int_{\Omega}|\nabla a|^2dx$$

subject to the PDE.

3.3 Wave speed inversion

- Forward problem:

$$\begin{cases} u_{tt} - \nabla(c^2(x)\nabla u) = f & \text{on } \Omega\times[0,T] \\ c^2\nabla u\cdot\hat{n} = 0 & \text{on } \partial\Omega\times[0,T] \\ u = u_t = 0 & \text{on } \Omega\times\{0\}. \end{cases}$$

- With Tikhonov-H^1 regularization, the inverse problem is

$$\min_{c(x)} \frac{1}{2}\int_0^T\int_{\Omega}(u-u_d)^2dxdt + \frac{\alpha}{2}\int_{\Omega}|\nabla c|^2dx$$

4 PDE-constrained optimization

- Given a minimization problem of some $\pi(u)$ over functions $u(x)$, look for a u that satisfies the Euler-Lagrange equations,

$$\left.\frac{\partial\pi(u+\epsilon\hat{u})}{\partial\epsilon}\right|_{\epsilon=0} = 0$$

for all $\hat{u}\in H_0^1(\Omega)$.

- Procedure:
  - Substitute $u+\epsilon\hat{u}$ into $\pi(u)$
  - Ignore ϵ^2 terms

- Take ϵ derivative, set $\epsilon = 0$, set expression to 0 to get weak form of Euler-Lagrange equations
- Express all terms as integrals against $\hat{u}$, using integration by parts to deal with $\nabla\hat{u}$ terms
- Since equation holds for all $\hat{u}$, whatever is being integrated against $\hat{u}$ must be 0. Gives strong form of Euler-Lagrange equations.

- Green's identity for integration-by-parts:

$$\int_\Omega k\nabla u \cdot \nabla \hat{u}\, dx = -\int_\Omega \nabla \cdot k\, \nabla u\, \hat{u}\, dx + \int_{\partial\Omega} k\, \nabla u \cdot \hat{n}\, \hat{u}\, dS.$$

- If the energy being minimized is quadratic, the Euler-Lagrange PDE is linear.

- Optimize-then-discretize (OTD, Galerkin): Find E-L equations in weak form and then substitute in finite element discretizations of u and $\hat{u}$, or solve strong E-L equations.

- Discretize-then-optimize (DTO): for a quadratic energy, plug in finite elements to get some $\pi(u_h) = \frac{1}{2}u_h^T K u_h + f^T u_h$, then solve $Ku_h = f$ for the optimizer.

- OTD and DTO are typically the same for linear problems if the discretization is done the same way

- For non-quadratic energy, the E-L equations are nonlinear, so need to solve them using a Newton iteration. The Newton step requires the Hessian, i.e., second variation:

$$\delta_u^2 \pi(u)(\tilde{u}, \hat{u}) = \left.\frac{\partial \delta_u \pi(u + \epsilon\tilde{u})(\hat{u})}{\partial \epsilon}\right|_{\epsilon=0},$$

  where

$$\delta_u \pi(u)(\hat{u}) = \left.\frac{\partial \pi(u + \epsilon\hat{u})}{\partial \epsilon}\right|_{\epsilon=0}$$

  is the first variation and $\tilde{u}$ is added as a new variation direction while keeping $\hat{u}$ constant.

- Then the Newton iteration is $u^{k+1} = u^k + \gamma\tilde{u}$ for $\tilde{u}$ the solution to

$$\delta_u^2 \pi(u^k)(\tilde{u}, \hat{u}) = -\delta_u \pi(u^k)(\hat{u}) \quad \text{for all } \hat{u} \in H_0^1(\Omega).$$

  In the weak form, this is an equation for $\tilde{u}$ that depends on u^k and $\hat{u}$, and must hold for all $\hat{u}$. In the strong form we get rid of $\hat{u}$. Can add a line search step and a modification of the Hessian to ensure it is positive definite to complete the method.

5 Sensitivity Analysis and Adjoint Equations

5.1 Finite-dimensional case

- Given a minimization problem

$$\min_m f(u(m), m) \quad \text{subject to } A(m)u = b,$$

  for some operator $A(m)$, with $u \in \mathbb{R}^M, m \in \mathbb{R}^N$, we want to find the gradient of f w.r.t. m to use a descent or Newton method.

- Sensitivity method:
  - One component of the gradient is

$$\frac{df}{dm_i} = \frac{\partial f}{\partial m_i} + \frac{\partial f}{\partial u}\frac{du}{dm_i},$$

    and noting that

$$\begin{aligned} \frac{\partial}{\partial m_i}(A(m)u(m) - b) = 0 &\implies A(m)\frac{du}{dm_i} + \frac{dA(m)}{dm_i}u(m) = 0 \\ &\implies A(m)\frac{du}{dm_i} = -\left(\frac{dA(m)}{dm_i}u(m)\right). \end{aligned}$$

  - Combining the m_i's into one equation gives the sensitivity equation,

$$A(m)\frac{du}{dm} = -\frac{d(A(m)u(m))}{dm},$$

    where the RHS denotes the $M \times N$ matrix with columns $\frac{dA(m)}{dm_i}u(m)$.
  - The direct sensitivity approach is to 1) Solve the forward problem $A(m)u = b$ (1 linear solve), 2) solve the sensitivity equation for the $M \times N$ matrix $\frac{du}{dm}$ (N solves), and 3) evaluate the gradient

$$\underbrace{g(m)^T}_{1\times N} = \underbrace{\frac{\partial f}{\partial m}}_{1\times N} + \underbrace{\frac{\partial f}{\partial u}}_{1\times M}\underbrace{\frac{du}{dm}}_{M\times N}$$

  - Can speed up by precomputing LU factorization of $A(m)$ and a preconditioner and reusing for the N solves.

- Adjoint sensitivity method
  - Instead, we could plug the sensitivity equations directly into the gradient expression to get

$$\underbrace{g(m)^T}_{1\times N} = \underbrace{\frac{\partial f}{\partial m}}_{1\times N} - \underbrace{\underbrace{\frac{\partial f}{\partial u}}_{1\times M}\underbrace{A(m)^{-1}}_{M\times M}}_{p^T}\underbrace{\frac{d(A(m)u(m))}{dm}}_{M\times N},$$

    where we have defined the adjoint variable

$$p^T = \frac{\partial f}{\partial u}A(m)^{-1},$$

    which therefore satisfies the adjoint equation

$$A(m)^T p = \left(\frac{\partial f}{\partial u}\right)^T.$$

  - So the adjoint method is 1) Solve the forward problem $A(m)u = b$, 2) Solve the adjoint equation for p, 3) evaluate the gradient

$$g(m)^T = \frac{\partial f}{\partial m} - p^T\frac{d(A(m)u)}{dm}.$$

    Requires only 2 solves, vs. $N + 1$ solves of sensitivity method.

- Lagrangian formulation of the adjoint method
  - Define the Lagrangian
$$L(u, m, p) = f(u, m) - p^T(A(m)u - b).$$
  - Then setting the derivatives w.r.t. p and u to 0 makes the derivative in m equal to the gradient:
$$\begin{aligned}
\frac{\partial L}{\partial p} &= A(m)u - b = 0 \quad \longrightarrow \quad \text{forward equation} \\
\frac{\partial L}{\partial u} &= \left(\frac{\partial f}{\partial u}\right)^T - A(m)^T p = 0 \quad \longrightarrow \quad \text{adjoint equation} \\
\frac{\partial L}{\partial m} &= \frac{\partial f}{\partial m} - p^T \frac{d(A(m)u)}{dm} = g(m)^T \quad \longrightarrow \quad \text{gradient.}
\end{aligned}$$
- Check gradients numerically after computing
- Exercise: recreate the example discrete-space calculation in the notes

5.2 Infinite dimensional case

- To apply the Lagrangian form of the adjoint method:
  - Write down the weak form of the constraint PDE, and construct a Lagrangian by adding taking the objective function and adding to it the weak form with $\hat{u}$ replaced by p.
  - Take a first variation in p with test function $\hat{p}$ to get the weak form of the forward/state equation.
  - Take a first variation in u to get a weak form of the adjoint equation.
  - Take a first variation in m with test function $\hat{m}$ to get a directional gradient in the $\hat{m}$ direction.
- For a time-dependent problem where $m(x)$ is the initial condition:
  - Integrate against $\hat{u}$ over both space Ω and time $[0, T]$ to get the weak form of the constraint PDE.
  - Form Lagrangian $L(u, m, p, q)$, as the sum of the objective function, p integrated against the constraint PDE, and q integrated against $u(0) - m$ to constrain the initial condition.
  - Taking a variation in $\hat{u}$ will give the adjoint equations, which in strong form include a condition for $p(T)$ and $q = p(0)$. The adjoint equations are solved backward in time.
  - After taking all the variations, the procedure to find the gradient in m at some m is 1) Solve the state equation for $u(m)$, 2) solve the adjoint equation backward in time for $p(x, t)$, 3) evaluate gradient using p and q in terms of p.

5.3 Computing the Hessian

- Idea is to built a new Lagrangian L^H where the objective function is the directional derivative in some arbitrary $\tilde{m}$ direction, and the constraints are the state and adjoint equations:

$$L^H(u, m, p; \tilde{u}, \tilde{m}, \tilde{p}) = \tilde{m}^T \left[\left(\frac{d(A(m)u)}{dm}\right)^T p + \frac{\partial f}{\partial m}\right] + \tilde{u}^T \left[A^T p + \left(\frac{\partial f}{\partial u}\right)^T\right] + \tilde{p}^T \left[A(m)u - b\right].$$

- As before we set the variations in $\tilde{p}$ and $\tilde{u}$ to 0:

$$
\begin{aligned}
\delta_p L^H(\hat{p}) = 0 &\implies A\tilde{u} - \frac{d(A(m)u)}{dm}\tilde{m} = 0 \qquad \text{incremental state equation} \\
\delta_u L^H(\hat{u}) = 0 &\implies A^T\tilde{p} = -L_{m,u}^T\tilde{m} - L_{u,u}\tilde{u} \qquad \text{incremental adjoint equation} \\
\delta_m L^H(\hat{m}) &= \underbrace{\left(L_{m,m}^T\tilde{m} + L_{u,m}^T\tilde{u} - \left(\frac{d(A(m)u)}{dm}\right)^T\tilde{p}\right)}_{H\tilde{m}}\hat{m} \qquad \text{Hessian,}
\end{aligned}
$$

  where $H\tilde{m}$ is the Hessian applied to $\tilde{m}$.

- To find Hessian applied to $\tilde{m}$, solve incremental state equation for $\tilde{u}$, then solve incremental adjoint equation for $\tilde{p}$, then plug into Hessian expression. Total of two matrix solves for one Hessian matvec.

- Same process in infinite dimensions.