Regularization and the "Adjoint Method" of Solving Inverse Problems

Patricia K. Lamm
Department of Mathematics
Michigan State University

(visiting:)
Department of Mathematics
The University of North Carolina at Chapel Hill

OUTLINE OF LECTURES

- **Lecture # 1**

  1. Contrast differences between the *finite dimensional* minimization problem,

  $$\min_{\beta \in \mathbb{R}^p} S(\beta), \quad \beta = (\beta_1, \ldots, \beta_p)^\top \in \mathbb{R}^p \quad .$$

     (where $\mathbb{R}^p$ is the usual p-dimensional Euclidean space), and the *infinite dimensional* minimization problem

  $$\min_{q \in F} S(q), \quad q = q(t) \in F$$

     (where F is a "function space").

  2. Methods for solving the minimization problem require computing $S'(\beta)$ (for the finite dimensional problem), or $S'(q)$ (for the infinite dimensional problem).
     (a) Computation of $S'(\beta)$
        i. Sensitivity equations
        ii. Matrix transpose (or adjoint)
     (b) Computation of $S'(q)$
        i. Sensitivity equations
        ii. Operator adjoint.

- **Lecture # 2**

  1. Methods for minimizing $S(\beta)$ (the finite dimensional problem):
     (a) Solving *necessary condition* equations, $S'(\beta) = 0$.
     (b) Descent Methods for minimizing $S(\beta)$ (these methods also use the derivative $S'(\beta)$).
        i. Steepest Descent Method
        ii. Conjugate Gradient Method

- **Lecture # 3**

  1. Methods for solving the *infinite dimensional* problem are generalizations of those required for the finite dimensional problem. The derivative $S'(q)$ is used in all methods.
  2. Implementation of Descent Methods for the *infinite dimensional* minimization problem, using both sensitivity equations and adjoint equations.
  3. Implementation of Descent Methods when regularization is present.

REFERENCES:

[1] J. V. Beck, B. Blackwell, and C. R. Clair, Jr.. *Inverse Heat Conduction.* Wiley-Interscience, 1985.

[2] J. L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations.* Springer-Verlag, 1971.

[3] D. G. Luenberger. *Optimization by Vector Space Methods.* John Wiley & Sons, Inc., 1969.

Lecture #1

Example: Inverse Heat Conduction Problem

Consider the following partial differential equation for heat conduction (no radiation or convection):

$$
\begin{aligned}
\rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\
T(x,0) &= T_0(x) \\
kT_x|_{x=L} &= f(t) \quad \text{(known boundary condition)} \\
-kT_x|_{x=0} &= q(t) \quad \text{(unknown boundary condition)}
\end{aligned}
$$

Here $T = T(x,t;q)$ denotes temperature, corresponding to a particular value of the *unknown* function q; the remaining entities, $\rho =$ density, $c =$ specific heat, and $k =$ thermal conductivity, are assumed to be *known* functions of x and t.

Without loss of generality, we will take $T_0(x) = 0$ and $f(t) = 0$; that is, T solves

$$
\begin{aligned}
\rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\
T(x,0) &= 0 \\
kT_x|_{x=L} &= 0 \\
-kT_x|_{x=0} &= q(t).
\end{aligned}
$$

These equations will be called the IHCP (inverse heat conduction problem) equations.

We will assume that there is one sensor located at position $x = d$, $0 < x < L$, for which temperature observations Y_i (corresponding to $T(x{=}d, t_i; q)$) are available at times t_i, $0 \leq t_i \leq t_{\max}$, for $i = 1, 2, \ldots, n$.

Estimation of $q = q(t)$

Ideal situation: Find $q(t)$ such that

$$Y_i = T(d, t_i; q), \quad i = 1, \ldots, n.$$

Realistic situation: Because we do not expect to exactly match the model solution T to measured data, we instead attempt to find $q(t)$ such that q minimizes the "fit-to-data criterion"

$$
\begin{aligned}
S(q) &= \frac{1}{2}\sum_{i=1}^{n} |Y_i - T(d, t_i; q)|^2 \\
&= (Y - T(d,\cdot;q))^{\mathsf{T}}(Y - T(d,\cdot;q))
\end{aligned}
$$

where $Y = (Y_1, \ldots, Y_n)^{\mathsf{T}}$ and $T(d,\cdot;q) = (T(d,t_1;q), \ldots, T(d,t_n;q))^{\mathsf{T}}$ are vectors in $\mathbb{R}^n$ and "T" denotes "transpose". We note that many other fit-to-data criteria (e.g., Maximum Likelihood, etc.) may be used with the methods discussed below.

Finite dimensional vs *infinite* dimensional minimization problems:

We will characterize the dimensionality of the minimization problem according to the dimension of the "parameter space" over which the minimization occurs. For example, suppose we assume an *a priori* representation for the unknown function $q(t)$, such as:

- (polynomial representation)

$$q(t) = \beta_1 + \beta_2 t + \beta_3 t^2 + \ldots + \beta_p t^{p-1}$$

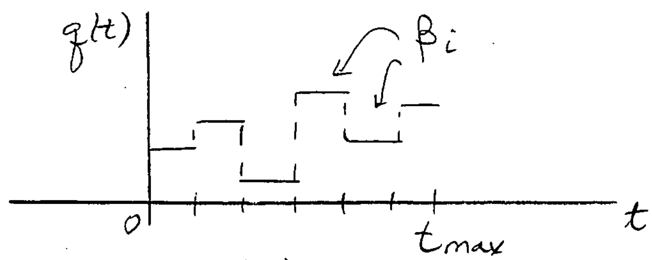
- (exponential representation)

$$q(t) = \beta_1 e^{\beta_2 t} + \beta_3 e^{\beta_4 t} + \ldots \beta_{p-1} e^{\beta_p}$$

- (step-function representation)

$$q(t) = \beta_i \quad \text{for } t_{i-1} \le t \le t_i, \quad i = 1, \ldots, p,$$

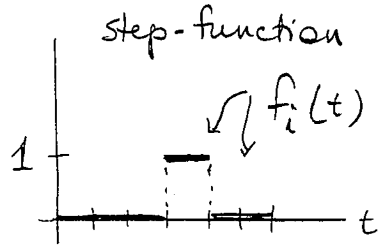
where $0 = t_0 < t_1 < \ldots < t_{p-1} < t_p = t_{\max}$,

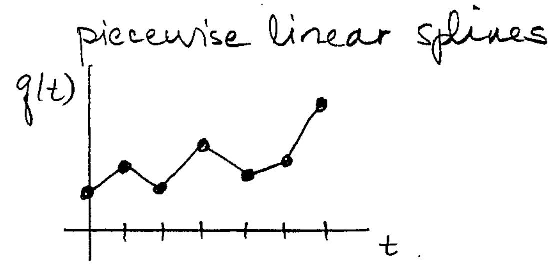


- (or, a general linear representation),

$$q(t) = \beta_1 f_1(t) + \beta_2 f_2(t) + \ldots + \beta_p f_p(t),$$

where $f_1(t), \ldots, f_p(t)$ are known, fixed functions of t.



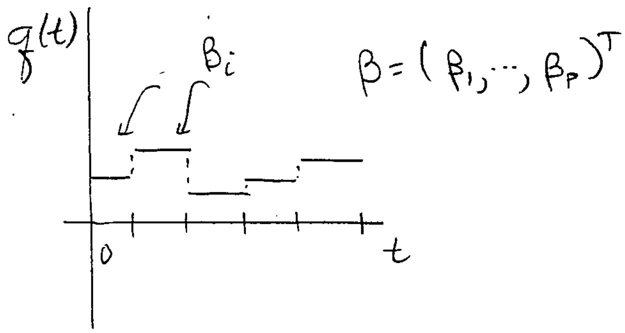


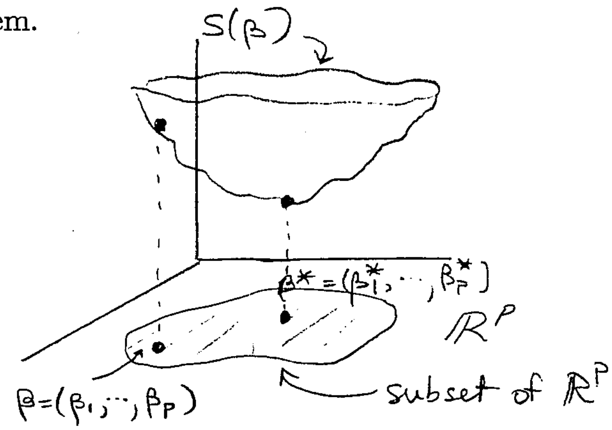
We note that, with the exception of the exponential representation for q, each of these representations has the property that the resulting function q is *linear* in the parameters $\beta_1, \beta_2, \ldots, \beta_p$.

In each of the above cases, the unknown function q has been parametrized in such a way that we are no longer minimizing over all possible *functions* q, but rather are minimizing over all possible *constants* $\beta_1, \beta_2, \ldots, \beta_p$. The "parameter space" is then p-dimensional, with unknown "parameters" given by the p-dimensional vectors of form $\beta \equiv (\beta_1, \beta_2, \ldots, \beta_p)^\top$ in $\mathbb{R}^p$. The corresponding minimization problem becomes

$$\min_{\beta \text{ in } \mathbb{R}^p} S(\beta) = \min_{\beta \text{ in } \mathbb{R}^p} \left\{ \frac{1}{2} \sum_{i=1}^{n} |Y_i - T(d, t_i; \beta)|^2 \right\}$$

and is considered a *finite dimensional* problem.





In contrast, the *infinite dimensional* minimization problem occurs when we do *not* select an *a priori* parametrization for the unknown function q. For example, we might consider q as belonging to a more general "function space" F, where

$$\begin{aligned} F &= L_2(0, t_{\max}) \\ &= \text{all "square integrable functions" defined on } [0, t_{\max}] \\ &= \{q(t) \text{ satisfying } \int_0^{t_{\max}} (q(t))^2 \, dt < \infty\}. \end{aligned}$$

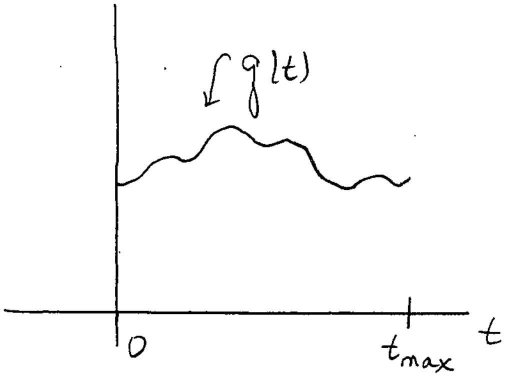
We would then minimize over this entire *function* space, an *infinite dimensional* space. The minimization then becomes

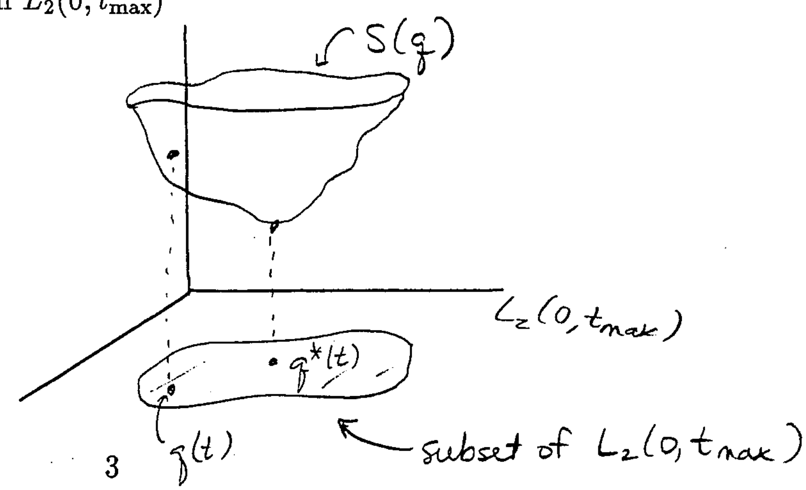
$$\min_{q \text{ in } L_2(0, t_{\max})} S(q) = \min_{q \text{ in } L_2(0, t_{\max})} \left\{ \frac{1}{2} \sum_{i=1}^{n} |Y_i - T(d, t_i; q)|^2 \right\},$$

an *infinite dimensional* minimization problem.

To facilitate subsequent calculations, we will assume (for the infinite dimensional case only) that we have "infinite observations" (i.e., measurements $Y(t)$ are available for all t in $(0, t_{\max})$), and that the minimization problem is given by the generalization

$$\min_{q \text{ in } L_2(0, t_{\max})} S(q) = \min_{q \text{ in } L_2(0, t_{\max})} \left\{ \frac{1}{2} \int_0^{t_{\max}} |Y(t) - T(d, t; q)|^2 \, dt \right\}.$$





Remarks Concerning Finite Dimensional vs Infinite Dimensional Minimization Problems

1. **Instability:** We know for finite dimensional problems that, as p (the number of parameters) increases, the inverse problem becomes more unstable and some type of regularization scheme must be invoked. Similarly, in the infinite dimensional problem (where we are essentially letting p go to infinity), it is not at all surprising that such problems are also often very "ill-posed", requiring regularization.

2. **Linearity:** Because the unknown function q appears as a boundary condition in the IHCP equations, the calculations that follow are greatly simplified. This is due to the fact that the solution T to the IHCP equations is *linear* in q; that is, from the principle of linear superposition, we know that $T(t,x;\alpha q_1 + \gamma q_2) = \alpha T(t,x;q_1) + \gamma T(t,x;q_2)$ where $T(x,t;q_i)$ is the solution to the IHCP equations with boundary condition q_i at $x = 0$ for $i = 1,2$.

   Similarly, for the finite dimensional problem, if a *linear representation* is chosen for q, such as the general form

   $$q(t) = \beta_1 f_1(t) + \beta_2 f_2(t) + \ldots + \beta_p f_p(t),$$

   it then follows that the solution to the IHCP equations is linear in β; in fact, for $T(x,t;\beta)$ the solution to the IHCP equation with this q appearing in the $x = 0$ boundary condition,

   $$\begin{aligned} \rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\ T(x,0) &= 0 \\ kT_x|_{x=L} &= 0 \\ -kT_x|_{x=0} &= \beta_1 f_1(t) + \beta_2 f_2(t) + \ldots + \beta_p f_p(t), \end{aligned}$$

   we have that this solution satisfies

   $$T(x,t;\beta) = \beta_1 T(x,t;f_1) + \cdots + \beta_p T(x,t;f_p),$$

   where $T(x,t;f_k)$ satisfies the IHCP equations with boundary condition $f_k(t)$ at $x = 0$:

   $$\begin{aligned} \rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\ T(x,0) &= 0 \\ kT_x|_{x=L} &= 0 \\ -kT_x|_{x=0} &= f_k(t). \end{aligned}$$

   — known, or presupposed, function e.g. step fcn

In fact, for a given β, the predicted temperature T at measurement location $x = d$ and times t_i, $i = 1, \ldots, n$, may be written as

$$
\begin{aligned}
T(d,\cdot;\beta) &\equiv \begin{pmatrix} T(d,t_1;\beta) \\ T(d,t_2;\beta) \\ \cdot \\ \cdot \\ \cdot \\ T(d,t_n;\beta) \end{pmatrix} = \begin{pmatrix} \beta_1 T(d,t_1;f_1) + \cdots + \beta_p T(d,t_1;f_p) \\ \beta_1 T(d,t_2;f_1) + \cdots + \beta_p T(d,t_2;f_p) \\ \cdot \\ \cdot \\ \cdot \\ \beta_1 T(d,t_n;f_1) + \cdots + \beta_p T(d,t_n;f_p) \end{pmatrix} \\
&= \begin{pmatrix} T(d,t_1;f_1) & \cdots & T(d,t_1;f_p) \\ T(d,t_2;f_1) & \cdots & T(d,t_2;f_p) \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ T(d,t_n;f_1) & \cdots & T(d,t_n;f_p) \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \cdot \\ \cdot \\ \cdot \\ \beta_p \end{pmatrix} \\
&\equiv X\beta,
\end{aligned}
$$

where X is the $n \times p$ constant-valued matrix ("sensitivity matrix") of solutions to the IHCP equations for given times t_i and given (known) boundary conditions $f_k(t)$.

The linearity of T in q will be useful in what follows. We note however that a similar theory holds for the case when T is *not* linear in the unknown parameter, as occurs for example in the case of unknown coefficients k, c, and ρ.

3. **"Quadratic" Fit-to-Data Criterion:** Because T is linear in q, it is easy to see that the "fit-to-data" criterion $S(q)$ (for the infinite dimensional problem),

$$S(q) = \frac{1}{2} \int_0^{t_{\max}} |Y(t) - T(d,t;q)|^2 \, dt,$$

is "quadratic" in q and that $S(\beta)$ (for the finite dimensional problem),

$$
\begin{aligned}
S(\beta) &= \frac{1}{2} \sum_{i=1}^{n} |Y_i - T(d,t_i;\beta)|^2 \\
&= \frac{1}{2} (Y - T(d,\cdot;\beta))^\top (Y - T(d,\cdot;\beta)) \\
&= \frac{1}{2} (Y - X\beta)^\top (Y - X\beta)
\end{aligned}
$$

is "quadratic" in β. This fact will be important when descent methods are discussed.

both "Hilbert Spaces"

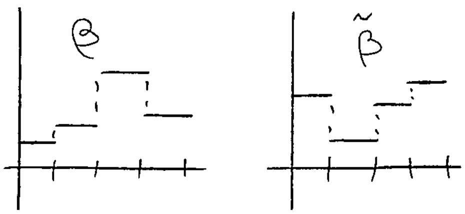
Finite dimensional parameter space = $\mathbb{R}^p$

Notion of multiplication

"Product" of $\mathbb{R}^p$-vectors β, $\tilde{\beta}$:

$$\langle \beta, \tilde{\beta} \rangle = \beta^\top \tilde{\beta}$$

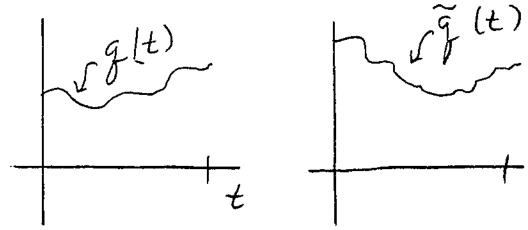
$$= \sum_{i=1}^{p} \beta_i \tilde{\beta}_i$$

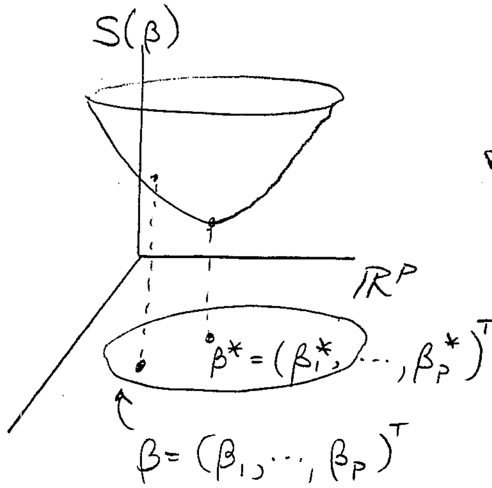


Infinite dimensional parameter space = $L_2(0, t_{\max})$

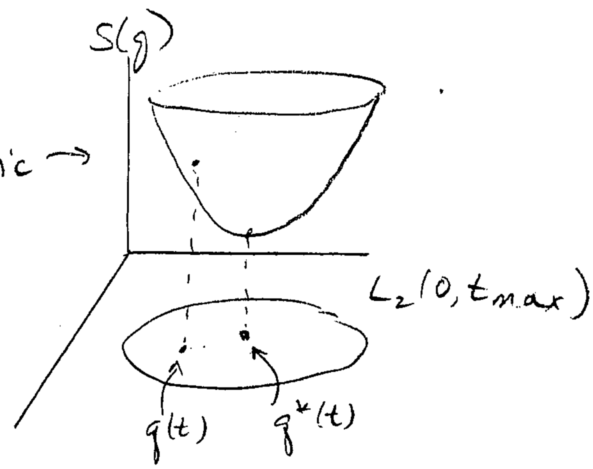
"Product" of $L_2(0, t_{\max})$-functions $q(t)$, $\tilde{q}(t)$:

$$\langle q, \tilde{q} \rangle = \int_0^{t_{\max}} q(t)\tilde{q}(t)\, dt$$





← quadratic →



Minimization Techniques:

(1) Set $S'(\beta) = 0$ + solve for β^*

or: (2) Use $-S'(\beta)$ to point "down" on surface

Descent method

(1) Set $S'(q) = 0$ + solve for q^*

or: (2) Use $-S'(q)$ to point "down" on surface.

$\Rightarrow$ Need derivatives $S'(\beta)$, $S'(q)$.

Computation of the Derivatives $S'(\beta)$, $S'(q)$

Schemes to minimize $S(\beta)$ (or $S(q)$) typically require one or more calculations of the derivative of S with respect to the unknown parameter; i.e., $S'(\beta)$ is required for the finite dimensional problem, and $S'(q)$ for the infinite dimensional problem.

First (Standard) Computation of $S'(\beta)$: (finite dimensional problem)

In the finite dimensional case, the derivative is just the usual gradient of S,

$$S'(\beta) = \nabla_\beta S \equiv \left(\frac{\partial S}{\partial \beta_1}, \frac{\partial S}{\partial \beta_2}, \ldots, \frac{\partial S}{\partial \beta_p}\right)^\top .$$

Thus, for

$$S(\beta) = \frac{1}{2}(Y - X\beta)^\top (Y - X\beta)$$

we have

$$\begin{aligned} S'(\beta) &= \frac{1}{2}\nabla_\beta \left[(Y - X\beta)^\top (Y - X\beta)\right] \\ &= \left[\nabla_\beta (Y - X\beta)^\top\right][Y - X\beta] \\ &= -X^\top [Y - X\beta] \end{aligned}$$

chain rule

transpose of sensitivity matrix

On the other hand, for the infinite dimensional problem, the derivative $S'(q)$ denotes the derivative of S with respect to a *function* q; although the correct interpretation of this derivative requires a good deal more mathematics (and thus its calculation is considerably more complex), we can at least attempt a *formal* understanding of $S'(q)$ as a generalization of a standard derivative.

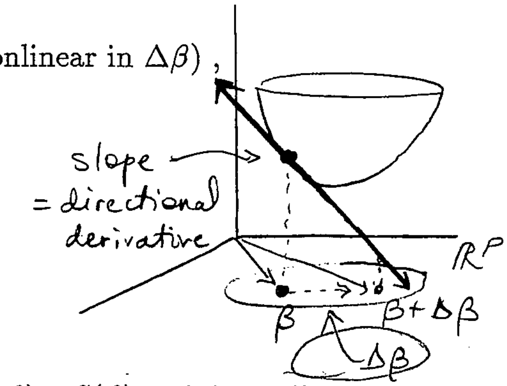
Second Computation of $S'(\beta)$: (for which $S'(q)$ is a generalization:)

Recall, from Taylor Series theory, that if h is a (sufficiently smooth) real–valued function of x, then for Δx in $\mathbb{R}$,

$$h(x + \Delta x) - h(x) = \underbrace{h'(x) \cdot \Delta x}_{\text{linear in } \Delta x} + \underbrace{\frac{1}{2}h''(x) \cdot \Delta x^2 + \ldots .}_{\text{terms nonlinear in } \Delta x}$$

Similarly, for β in $\mathbb{R}^p$, and $\Delta\beta$ in $\mathbb{R}^p$, the *"directional derivative of S at β in the direction of $\Delta\beta$"* is given by $D_{\Delta\beta}S(\beta)$, which satisfies

$$\underbrace{S(\beta+\Delta\beta)-S(\beta)}_{\text{in } \mathbb{R}} = \underbrace{D_{\Delta\beta}S(\beta)}_{\substack{\text{linear in } \Delta\beta \\ =\text{directional} \\ \text{derivative}}} + \text{(terms nonlinear in } \Delta\beta\text{)},$$



where each term in the above expression is a real number.

To calculate this directional derivative, we compute $S(\beta+\Delta\beta)-S(\beta)$ and drop all terms which are nonlinear in $\Delta\beta$:

$$\begin{aligned} S(\beta+\Delta\beta)-S(\beta) &= \frac{1}{2}(Y-X(\beta+\Delta\beta))^\top(Y-X(\beta+\Delta\beta)) - \frac{1}{2}(Y-X\beta)^\top(Y-X\beta) \\ &= \frac{1}{2}([Y-X\beta]-X\Delta\beta)^\top([Y-X\beta]-X\Delta\beta) - \frac{1}{2}(Y-X\beta)^\top(Y-X\beta) \\ &= \frac{1}{2}(Y-X\beta)^\top(-X\Delta\beta) + \frac{1}{2}\left(-X\Delta\beta)^\top(Y-X\beta\right) + \frac{1}{2}(-X\Delta\beta)^\top(-X\Delta\beta). \end{aligned}$$

But $(Y-X\beta)^\top(-X\Delta\beta)$ belongs to $\mathbb{R}$ so

$$[(Y-X\beta)^\top(-X\Delta\beta)]^\top = (Y-X\beta)^\top(-X\Delta\beta).$$

Thus,

$$S(\beta+\Delta\beta)-S(\beta) = \underbrace{(Y-X\beta)^\top(-X\Delta\beta)}_{\text{term linear in } \Delta\beta} + \underbrace{\frac{1}{2}(-X\Delta\beta)^\top(-X\Delta\beta)}_{\text{term nonlinear in } \Delta\beta}.$$

Therefore, the needed *directional derivative* is the *linear* term in the Taylor expansion, or

$$D_{\Delta\beta}S(\beta) = (Y-X\beta)^\top(-X\Delta\beta).$$

(Compare this with $S'(\beta) = \nabla_\beta S$ found earlier, $S'(\beta) = -X^\top(Y-X\beta)$.)

Remark: We note that we could have used the fact that $T(d,\cdot;\beta) = X\beta$ to rewrite the above as

$$\begin{aligned} S(\beta+\Delta\beta)-S(\beta) &\equiv \frac{1}{2}(Y-T(d,\cdot;\beta+\Delta\beta))^\top(Y-T(d,\cdot;\beta+\Delta\beta)) \\ &\quad -\frac{1}{2}(Y-T(d,\cdot;\beta))^\top(Y-T(d,\cdot;\beta) \\ &= \underbrace{(Y-T(d,\cdot;\beta))^\top(-D_{\Delta\beta}T(d,t))}_{\text{term linear in } \Delta\beta} \\ &\quad + \underbrace{\frac{1}{2}(-D_{\Delta\beta}T(d,t))^\top(-D_{\Delta\beta}T(d,t))}_{\text{term nonlinear in } \Delta\beta}. \end{aligned}$$

We thus have that the directional derivative of S at β in the direction of $\Delta\beta$ is given, as above, by

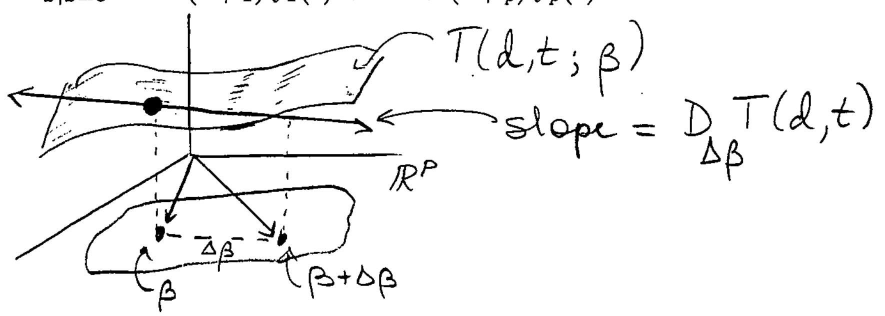
$$D_{\Delta\beta}S(\beta) = (Y - X\beta)^{\top}(-X\Delta\beta)$$

or, equivalently,

$$D_{\Delta\beta}S(\beta) = \underbrace{(Y - T(d,\cdot;\beta))^{\top}}_{\text{residual}}\underbrace{(-D_{\Delta\beta}T(d,t))}_{\substack{\text{directional}\\ \text{derivative}\\ \text{of T at }\beta}}.$$

In this case, $D_{\Delta\beta}T(d,t) = X\Delta\beta$ is interpreted as the directional derivative of T at β in the direction of $\Delta\beta$. In fact, $D_{\Delta\beta}T(x,t) \equiv \theta(x,t,;\Delta\beta)$, where θ satisfies the *sensitivity equations*

$$\begin{aligned}
\rho c\frac{\partial\theta}{\partial t} &= \frac{\partial}{\partial x}\left(k\frac{\partial\theta}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L\\
\theta(x,0) &= 0\\
k\theta_x|_{x=L} &= 0\\
-k\theta_x|_{x=0} &= (\Delta\beta_1)\, f_1(t) + \ldots + (\Delta\beta_p)\, f_p(t).
\end{aligned}$$



Question: How do we use the ***directional derivative*** $D_{\Delta\beta}S(\beta)$ **to find the** ***ordinary derivative*** $S'(\beta)$**?**

We recall that in $\mathbb{R}^p$, the *directional derivative* of any function $f(\beta)$ at β in the direction of (a unit vector) $\Delta\beta$ is always given by

$$D_{\Delta\beta}f(\beta) = \underbrace{(\nabla_\beta f(\beta))}_{\text{ordinary derivative}}{}^{\top}(\Delta\beta).$$

Therefore, for the function $S(\beta)$,

$$\begin{aligned}
D_{\Delta\beta}S(\beta) &= (\nabla_\beta S(\beta))^{\top}(\Delta\beta)\\
&= \langle \underbrace{\nabla_\beta S(\beta)}_{\substack{\text{vector}\\ \text{in } \mathbb{R}^p\\ = S'(\beta)}},\ \underbrace{\Delta\beta}_{\substack{\text{the}\\ \text{unit}\\ \text{direction}}} \rangle,
\end{aligned}$$

dot or scalar product

where $\langle\cdot\,,\,\cdot\rangle$ is the $\mathbb{R}^p$ scalar product.

Using this fact to "recover" $S'(\beta) \equiv \nabla_\beta S$ from $D_{\Delta\beta} S(\beta)$, we have, from above,

$$\begin{aligned} D_{\Delta\beta} S(\beta) &= (Y - T(d,\cdot;\beta))^\top \left(\underbrace{-D_{\Delta\beta} T(d,t)}_{=-X(\Delta\beta)} \right) \\ &= \left(-X^\top (Y - T(d,\cdot;\beta))\right)^\top \Delta\beta \\ &= \langle \underbrace{-X^\top (Y - T(d,\cdot;\beta))}_{\text{"derivative"}}, \underbrace{\Delta\beta}_{\text{"direction"}} \rangle. \end{aligned}$$

It thus follows that $S'(\beta) = -X^\top (Y - T(d,\cdot;\beta))$.

Summary of steps to find $D_{\Delta\beta} S(\beta)$ and $S'(\beta)$:

- Compute $S(\beta + \Delta\beta) - S(\beta)$ and drop terms nonlinear in $\Delta\beta$. What remains is the *directional derivative* $D_{\Delta\beta} S(\beta)$.

- Rewrite the *directional derivative* as

$$\begin{aligned} D_{\Delta\beta} S(\beta) &= (\text{"vector"})^\top \Delta\beta \\ &= \langle \text{"vector"}, \Delta\beta \rangle. \end{aligned}$$

  Once this is done, we identify:

$$S'(\beta) = \text{"vector"}.$$

  Thus, the directional derivative is rewritten

$$\begin{aligned} D_{\Delta\beta} S(\beta) &= (Y - T(d,\cdot;\beta)) \left(\begin{array}{c} \underbrace{-D_{\Delta\beta} T(d,t)} \\ = \\ \underbrace{-X} \Delta\beta \end{array} \right) \\ &= \left(\underbrace{-X^\top} (Y - T(d,\cdot;\beta)) \right)^\top (\Delta\beta) \\ &= \langle \underbrace{-X^\top} (Y - T(d,\cdot;\beta)), \Delta\beta \rangle, \end{aligned}$$

  and

$$S'(\beta) = -X^\top (Y - T(d,\cdot;\beta)).$$

Calculations required in order to compute $S'(\beta)$ for the IHCP, given a value of β:

To compute, for given $\beta = (\beta_1, \ldots, \beta_p)$,

$$S'(\beta) = -X^\top (Y - T(d, \cdot; \beta))$$

the following steps are taken:

- Solve the IHCP equations with the given β for the vector

$$T(d, \cdot; \beta) = (T(d, t_1; \beta), T(d, t_2; \beta), \ldots, T(d, t_n; \beta))^\top,$$

where $T(d, t_i; \beta)$ is the solution at $x = d$, $t = t_i$, of:

$$\begin{aligned} \rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\ T(x, 0) &= 0 \\ kT_x|_{x=L} &= 0 \\ -kT_x|_{x=0} &= \beta_1 f_1(t) + \beta_2 f_2(t) + \ldots + \beta_p f_p(t). \end{aligned}$$

Compute the *residual vector* $Y - T(d, \cdot; \beta)$.

- Multiply the *matrix transpose* (or *matrix adjoint*) of X times the residual $Y - T(d, \cdot; \beta)$. Here X is the "sensitivity matrix",

$$X = \begin{pmatrix} T(d, t_1; f_1) & \cdots & T(d, t_1; f_p) \\ T(d, t_2; f_1) & \cdots & T(d, t_2; f_p) \\ \vdots & \ddots & \vdots \\ T(d, t_n; f_1) & \cdots & T(d, t_n; f_p) \end{pmatrix},$$

and the entry $T(d, t_i; f_k)$ is the solution $\theta(x, t; f_k)$ at $x = d$, $t = t_i$, of the "sensitivity equations"

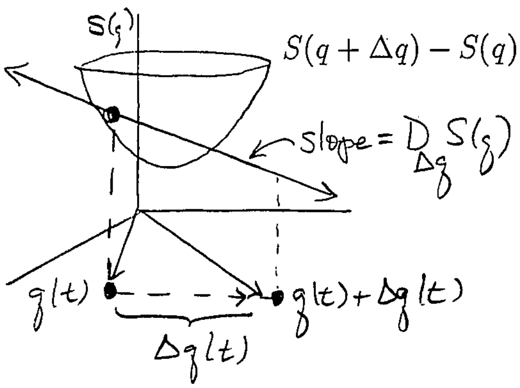
$$\begin{aligned} \rho c \frac{\partial \theta}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial \theta}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\ \theta(x, 0) &= 0 \\ k\theta_x|_{x=L} &= 0 \\ -k\theta_x|_{x=0} &= f_k(t). \end{aligned}$$

in this sense

Similar steps are taken to find the (infinite dimensional) directional derivative, $D_{\Delta q}S(q)$, and the (infinite dimensional) ordinary derivative, $S'(q)$:

- Compute $S(q+\Delta q) - S(q)$ (where q and Δq are given functions in $L_2(0, t_{\max})$) and drop terms nonlinear in Δq. What remains is the *directional* derivative $D_{\Delta q}S(q)$.

  Just as in the finite dimensional case, we compute

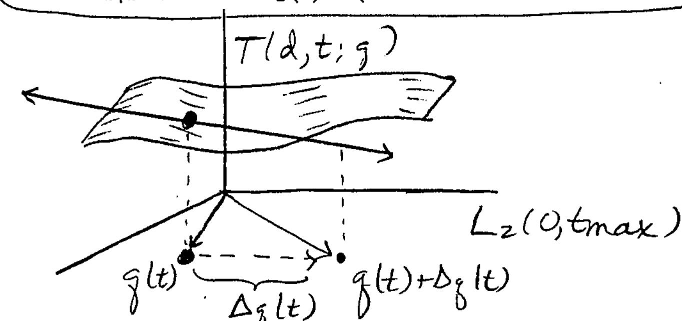


$$
\begin{aligned}
S(q+\Delta q) - S(q) \;&\equiv\; \frac{1}{2}\int_0^{t_{\max}} (Y(t) - T(d,t;q+\Delta q))\,(Y(t) - T(d,t;q+\Delta q))\,dt \\
&\quad - \frac{1}{2}\int_0^{t_{\max}} (Y(t) - T(d,t;q))\,(Y(t) - T(d,t;q))\,dt \\
&= \underbrace{\int_0^{t_{\max}} (Y(t) - T(d,t;q))(-D_{\Delta q}T(d,t))\,dt}_{\text{term linear in } \Delta q} \\
&\quad + \underbrace{\frac{1}{2}\int_0^{t_{\max}} (-D_{\Delta q}T(d,t))(-D_{\Delta q}T(d,t))\,dt,}_{\text{term nonlinear in } \Delta q}
\end{aligned}
$$

where $D_{\Delta q}T(d,t)$ is interpreted, as before, to be the directional derivative of T at q in the direction of Δq. As before, $D_{\Delta \beta}T(x,t)$ satisfies the sensitivity equations in θ,

$$
\begin{aligned}
\rho c \frac{\partial \theta}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial \theta}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\
\theta(x,0) &= 0 \\
k\theta_x|_{x=L} &= 0 \\
-k\theta_x|_{x=0} &= \Delta q(t) \quad \text{(the "direction function")}
\end{aligned}
$$

Same sense as before



We thus find, for the infinite dimensional case, that the *directional derivative of S at q in the direction of the function* Δq is given by

$$
D_{\Delta q}S(q) = \underbrace{\int_0^{t_{\max}} (Y(t) - T(d,t;q))}_{\text{residual}}\,\underbrace{(-D_{\Delta q}T(d,t))}_{\substack{\text{directional}\\ \text{derivative}\\ \text{of T at } q}}\,dt.
$$

- For the finite dimensional case, the next step in finding the derivative $S'(\beta)$ was to rewrite

$$\begin{aligned} D_{\Delta q} S(q) &= \langle \text{"vector"}, \Delta\beta \rangle \\ &= (\text{"vector"})^{\top} \Delta\beta \end{aligned}$$

and identify

$$S'(\beta) = \text{"vector"}.$$

recall this

Want to find derivative based on the directional derivative

In the infinite dimensional case, similar steps are required:

$$\begin{aligned} D_{\Delta q} S(q) &= \langle \text{"function of } t, \Delta q \rangle \\ &= \int_0^{t_{\max}} (\text{"function of t"}) \Delta q(t)\, dt. \end{aligned}$$

Once this is done, we identify the derivative

$$S'(q) = \text{"function of t"}.$$

But in the finite dimensional case, the "separation" of $\Delta\beta$ from the directional derivative $D_{\Delta\beta} T(d,t)$ was easy:

$$\begin{aligned} D_{\Delta\beta} S(\beta) &= (Y - T(d,\cdot;\beta))^{\top} \left(\underbrace{-D_{\Delta\beta} T(d,t)}_{=-X\Delta\beta} \right) \\ &= \underbrace{\left(\underbrace{-X^{\top}} (Y - T(d,\cdot;\beta))^{\top} \right)}_{S'(\beta)} (\Delta\beta) \end{aligned}$$

Things are not so straightforward in the infinite dimensional case. For this case,

$$D_{\Delta q} S(q) = \int_0^{t_{\max}} (Y(t) - T(d,t;q)) \left(\underbrace{-D_{\Delta q} T(d,t)}_{\neq \text{"matrix"} \cdot \Delta q} \right) dt$$

and that we cannot simply separate (via matrix transpose) the Δq from T's directional derivative $D_{\Delta q} T(d,t)$. However, we can (via several integration by parts) *construct a function of t* which makes the needed equality hold, namely,

"Sort of integration by parts"

$$\begin{aligned} D_{\Delta q} S(q) &\equiv \int_0^{t_{\max}} (Y(t) - T(d,t;q))(-D_{\Delta q} T(d,t))\, dt \\ &= \int_0^{t_{\max}} (\text{"function of t"}) \cdot \Delta q(t)\, dt. \end{aligned}$$

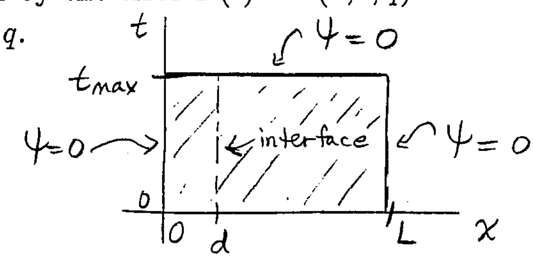
This function can be shown to be the solution ψ, evaluated at $x = 0$, of the following "adjoint equations:"

$$
\begin{aligned}
\frac{\partial}{\partial t}(c\rho\psi) &= -\frac{\partial}{\partial x}(k\frac{\partial \psi}{\partial x}) \quad 0 < t < t_{\max}, \quad 0 < x < L \\
\psi(x, t_{\max}) &= 0 \\
k\psi_x|_{x=L} &= 0 \\
-k\psi_x|_{x=0} &= 0 \\
\psi(d^+, t) - \psi(d^-, t) &= 0 \\
-\left(k\frac{\partial \psi}{\partial x}(d^+, t) - k\frac{\partial \psi}{\partial x}(d^-, t)\right) &= Y(t) - T(d, t; q).
\end{aligned}
$$

Final value problem

jump conditions

We note that the adjoint equations are solved *backwards in time* and have two *jump conditions* imposed at the "interface" $x = d$; the solution ψ of this system of equations is driven by the error $Y(t) - T(d, t; q)$ between measurements and model for a given value of q.



For comparison with the adjoint equations, we recall the original IHCP equations:

$$
\begin{aligned}
\rho c\frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\
T(x, 0) &= 0 \\
kT_x|_{x=L} &= 0 \\
-kT_x|_{x=0} &= q(t).
\end{aligned}
$$

Thus, for the infinite dimensional case, the derivative of $S(q)$ is given by

$$
S'(q) = \psi|_{x=0},
$$

a function of t.

Summary of calculations required in order to compute $S'(q)$ for a given value of q:

- Solve the IHCP equations, with the given q, for the function $T(d,t;q)$:

$$\begin{aligned}
\rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\
T(x,0) &= 0 \\
kT_x|_{x=L} &= 0 \\
-kT_x|_{x=0} &= q(t)
\end{aligned}$$

Compute the *residual error function* $Y(t) - T(d,t;q)$ corresponding to q.

- Compute the solution $\psi(x,t;q)$ of the adjoint equations:

$$\begin{aligned}
\frac{\partial}{\partial t}(c\rho\psi) &= -\frac{\partial}{\partial x}(k\frac{\partial \psi}{\partial x}) \quad 0 < t < t_{\max}, \quad 0 < x < L \\
\psi(x,t_{\max}) &= 0 \\
k\psi_x|_{x=L} &= 0 \\
-k\psi_x|_{x=0} &= 0 \\
\psi(d^+,t) - \psi(d^-,t) &= 0 \\
-\left(k\frac{\partial \psi}{\partial x}(d^+,t) - k\frac{\partial \psi}{\partial x}(d^-,t)\right) &= Y(t) - T(d,t;q).
\end{aligned}$$

(Ricatti Equations?)

- The derivative $S'(q)$ (a function of t) is given by $S'(q) = \psi|_{x=0}$.

$\psi \sim X^T$ "ADJOINT" or transpose

Lecture #2

Minimization Techniques: Necessary Conditions and Descent Methods

We will first consider the general problem of minimizing $S(\beta)$ over β in $\mathbb{R}^p$.

First Approach: Necessary Conditions for a Minimum.
Solve:

$$S'(\beta) = 0,$$

or

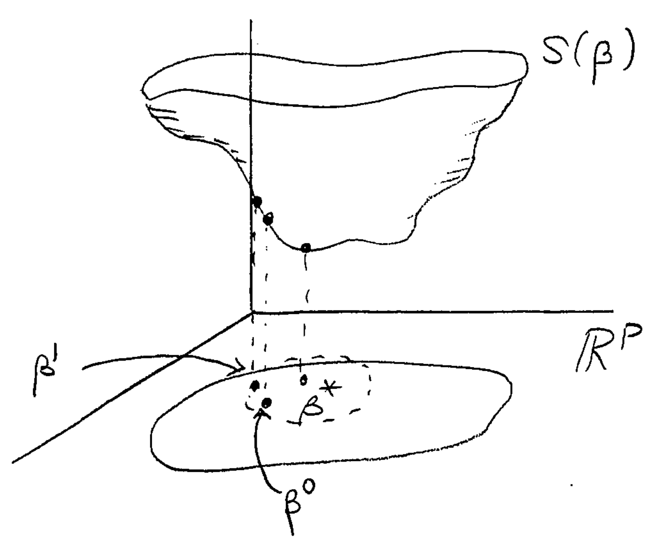
$$\begin{pmatrix} \frac{\partial S}{\partial \beta_1} \\ \vdots \\ \frac{\partial S}{\partial \beta_p} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix},$$

a set of p equations in the p unknowns $\beta_1, \ldots, \beta_p$. Solving this system for the β_i may be difficult, especially if these equations are nonlinear in the β_i. A numerical method (such as Newton's Method, or another iterative method) may be used, generating a sequence

$$\beta^0, \beta^1, \ldots, \beta^n, \ldots$$

where each β^n belongs to $\mathbb{R}^p$ for all n, and the $\mathbb{R}^p$-vector β^0 is a (user-supplied) initial guess.

Result for Newton's Method: Convergence of $\{\beta^n\}$ is guaranteed if β^0 is "close" to a local minimum β^*.

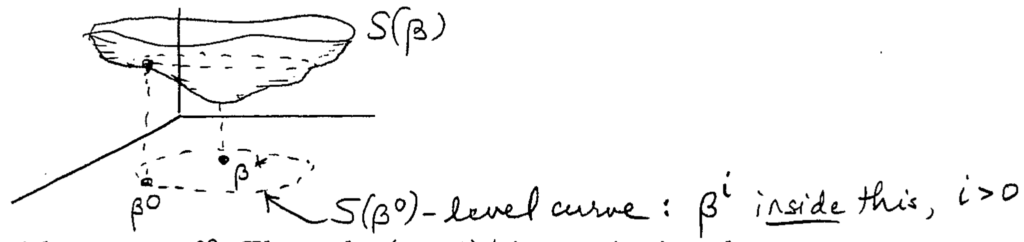


Second Approach: Descent Methods. These are also iterative techniques. Again, starting with an initial guess β^0, a sequence of iterates $\beta^1, \ldots, \beta^n, \ldots$ is generated in a manner determined by the particular descent method used.

General Approach for Descent Methods:
IDEA: Starting with β^0, pick the sequence $\beta_1, \beta_2, \ldots$ such that

$$S(\beta^0) \geq S(\beta^1) \geq S(\beta^2) \geq \ldots \geq S(\beta^n) \geq \ldots.$$

We note that this property of "descent of S" is *not* guaranteed by Newton's method.



Method: Start with a guess, β^0. Then, the $(n+1)^{\text{st}}$ iterate is given by

$$\beta^{n+1} = \beta^n + \alpha^n p^n \quad n = 0, 1, \ldots,$$

where

- β^n is the last iterate
- p^n is a particular "direction" vector in $\mathbb{R}^p$
- α^n is a real number chosen such that

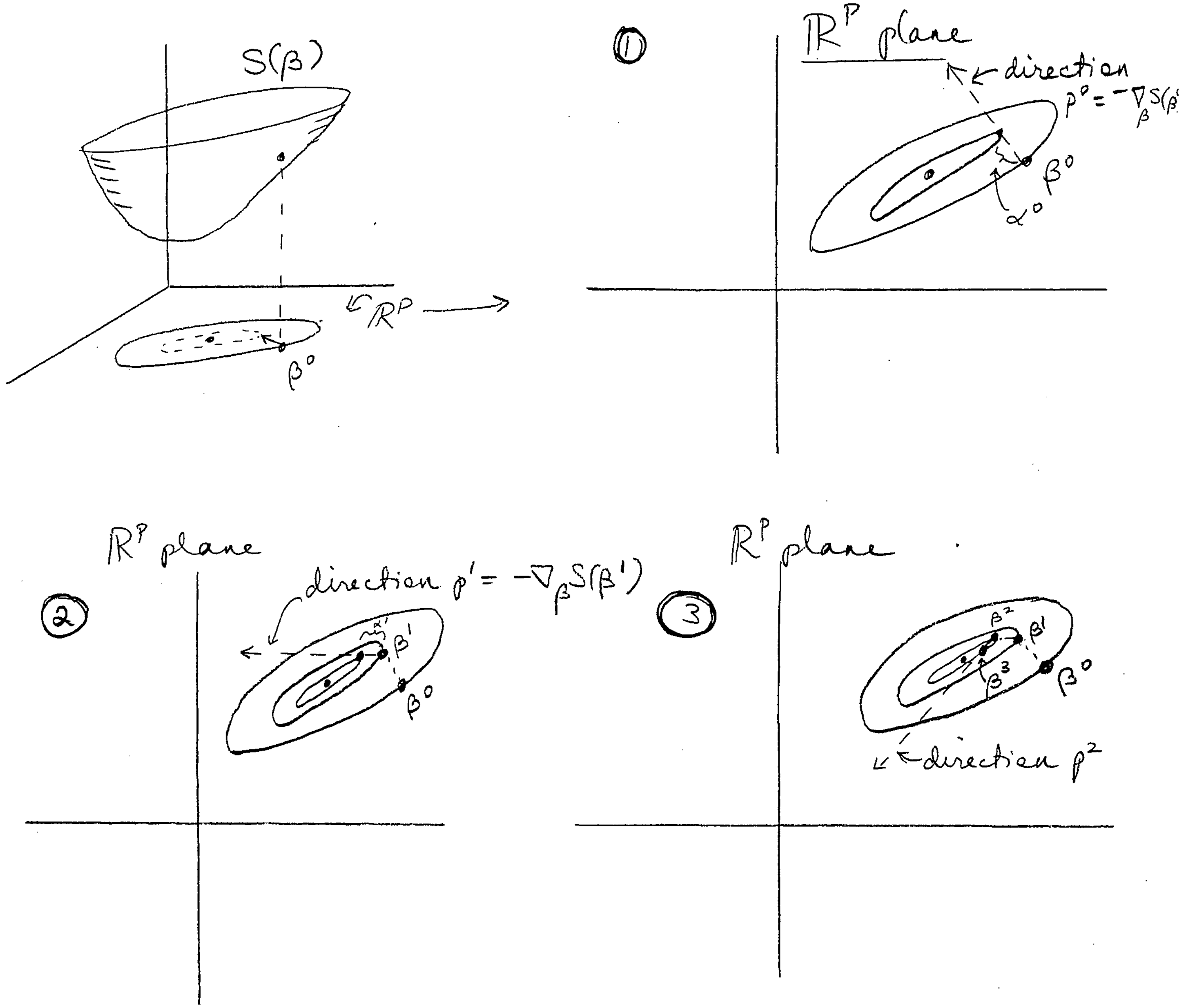
$$S(\beta^n) \geq S\underbrace{(\beta^n + \alpha^n p^n)}_{\beta^{n+1}}.$$

Method for choosing p^n:

- If $p^n = -(\text{gradient}) = -S'(\beta^n)$, this is the *Steepest Descent Method.*
- If $p^n =$ "orthogonal" (or "conjugate"), in a certain sense, to $p^0, p^1, \ldots, p^{n-1}$, (and p^0 is the initial gradient $-S'(\beta^0)$), this is the *Conjugate Gradient Method.*

Result for Descent Method: Convergence of $\{\beta^n\}$ to a local minimum β^* is guaranteed, for *any* initial guess β^0.

NOTE: Both methods are especially suited for *quadratic* problems (that is, when S is quadratic in β); in this case, the formulas are easy and the theory straightforward. A similar "local theory" is also possible in the case where S is not quadratic in β.


S(β)
$\mathbb{R}^p$
β⁰
①
$\mathbb{R}^p$ plane
direction
$p^0 = -\nabla_\beta S(\beta$
β⁰
α⁰
$\mathbb{R}^p$ plane
②
direction $p' = -\nabla_\beta S(\beta')$
α'
β'
β⁰
$\mathbb{R}^p$ plane
③
β²
β'
β⁰
β³
direction p^2

Method of Steepest Descent:

1. Pick an initial guess, β^0, for β in $\mathbb{R}^p$. Set $n = 0$.

2. Calculate $S'(\beta^n) = \nabla_\beta S(\beta^n)$ in $\mathbb{R}^p$. Set

$$p^n = -\nabla_\beta S(\beta^n).$$

3. Calculate α^n which satisfies the necessary condition for $S(\beta^n + \alpha p^n)$ to be minimized at α:

$$\frac{d}{d\alpha} S(\beta^n + \alpha p^n) = 0,$$

where β^n and p^n are given in previous steps.

4. Set

$$\beta^{n+1} = \beta^n + \alpha^n p^n.$$

5. If $\| \beta^{n+1} - \beta^n \|_{\mathbb{R}^p} < \epsilon$, stop. Otherwise, set $n = n + 1$ and go to step #2.

(Note: S must be "well-behaved" in order to guarantee our ability to perform the above steps; ideally, S is "nearly quadratic" and convex. Otherwise, some sort of positivity is needed for the matrix $S''(\beta)$.)

Implementation of Steepest Descent for $S(\beta) = \frac{1}{2}(Y - T(d, \cdot; \beta))^\top (Y - T(d, \cdot; \beta))$:

- **Selection of p^n:** We have already looked at the computation of $S'(\beta) = \nabla_\beta S(\beta)$ for the finite dimensional problem. For the Method of Steepest Descent, then,

$$\begin{aligned} p^n &= -\nabla_\beta S(\beta^n) \\ &= X^\top (Y - X\beta^n) \\ &= X^\top (Y - T(d, \cdot; \beta^n)). \end{aligned}$$

- **Selection of α^n:** We find that value of α which minimizes $S(\beta^n + \alpha p^n)$ (recall that β^n and p^n have already been given). That is, we find α such that

$$\begin{aligned} 0 &= \frac{d}{d\alpha} S(\beta^n + \alpha p^n) \\ &= \frac{d}{d\alpha} \left\{ \frac{1}{2} (Y - X(\beta^n + \alpha p^n))^\top (Y - X(\beta^n + \alpha p^n)) \right\} \\ &= (-Xp^n)^\top (Y - X(\beta^n + \alpha p^n)) \end{aligned}$$

or α^n solves

$$(Xp^n)^\top (Y - X\beta^n) = (-Xp^n)^\top \alpha^n (-Xp^n)$$

or

$$
\begin{aligned}
\alpha^n &= \frac{(Xp^n)^\top (Y - X\beta^n)}{(Xp^n)^\top (Xp^n)} \\
&= \frac{(p^n)^\top \left[X^\top (Y - X\beta^n)\right]}{(Xp^n)^\top (Xp^n)} \\
&= \frac{(p^n)^\top (p^n)}{(Xp^n)^\top (Xp^n)} \\
&= \frac{\|p^n\|^2}{\|Xp^n\|^2}
\end{aligned}
$$

where $\|\cdot\|$ denotes the usual Euclidean norm, $\|\beta\| = \beta^\top \beta$. Therefore,

$$\alpha^n = \frac{\|p^n\|^2}{\|D_{p^n} T(d,\cdot)\|^2}.$$

directional derivative of 'T' in the p^n direction

Under (reasonable) conditions on the original minimization problem, we have the following result for the Method of Steepest Descent:

Theorem: *For* any *starting point β^0, the sequence of iterates*

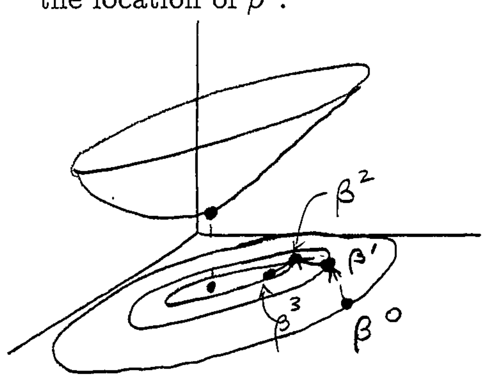
$$\beta^0, \beta^1, \ldots, \beta^n, \ldots$$

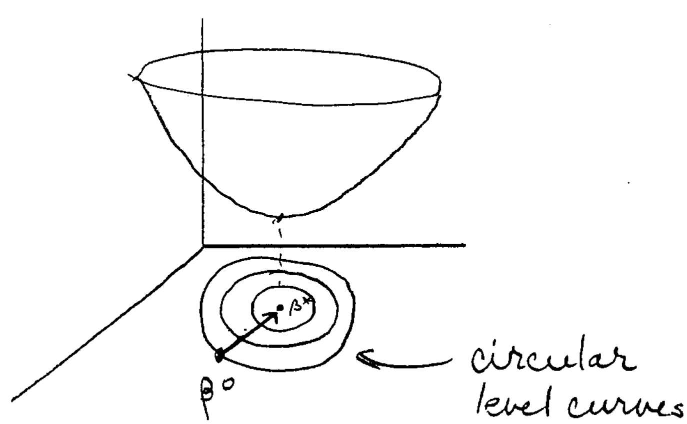
converges to a minimizer of $S(\beta)$ which closest to β^0. If the minimum is unique, β^n converges to the unique minimum β^*.

Remark: Not surprisingly, the speed of this convergence depends on:

- location of β^0,
- how **non**-circular the level curves are, e.g., eccentricity of the level-curve ellipses if in $\mathbb{R}^2$).

Further, if the level curves are circles (in $\mathbb{R}^2$), we get convergence in one step regardless of the location of β^0.

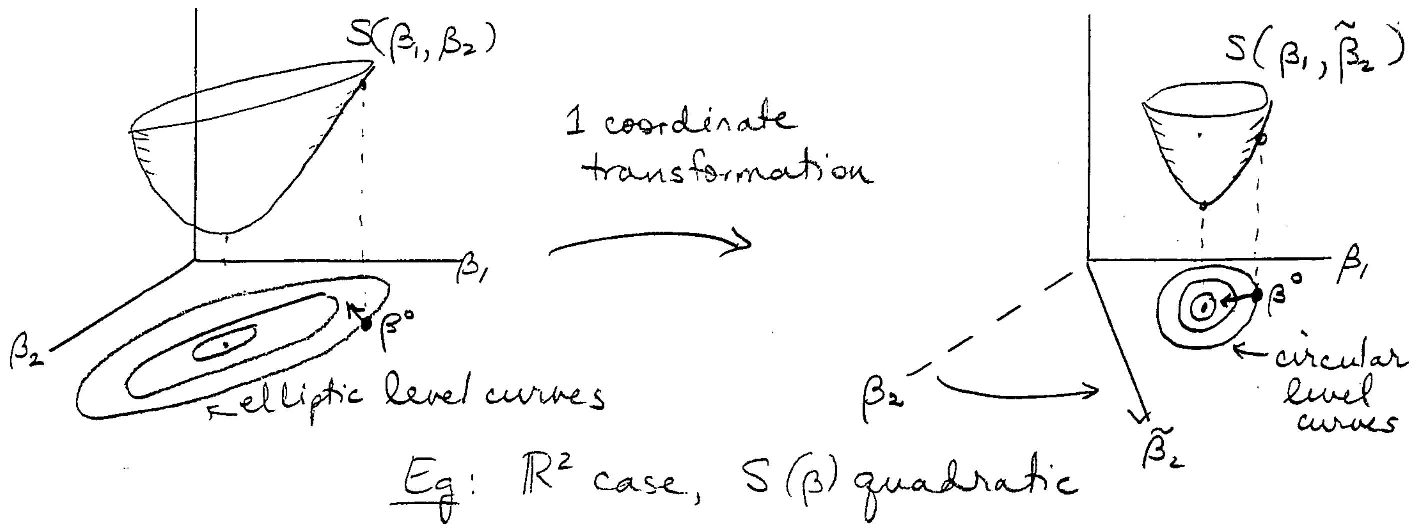




Conjugate Gradient Method: This method constructs a sequence of "orthogonal" or "conjugate" (in a certain sense) direction vectors $p^0, p^1, \ldots, p^n, \ldots$. The "stepsizes" $\alpha^1, \ldots, \alpha^n, \ldots$ are selected based on a particular Fourier series expansion.

IDEA: In some sense, the conjugate gradient method is based on the idea that $\mathbb{R}^p$, with the usual $\mathbb{R}^p$ scalar product, may *not* be the right "space" in which to minimize $S(\beta)$, since the level curves of $S(\beta)$ in this space may not be *circular*. In conjugate gradient, a new scalar product is used that amounts to a coordinate transformation of $\mathbb{R}^p$; implementation of this coordinate transformation takes at most $p-1$ steps, the resulting coordinate system being such that the level curves of $S(\beta)$ are *now circular*. It then takes **one** step to get the minimum of S (for a total of p steps).

Thus, the Method of Conjugate Gradients, though slightly more complicated to implement, generally converges *much faster* than the Method of Steepest Descent.



Formulas for p^n and α^n, in the case of the Conjugate Gradient Method:

1. Pick an initial guess β^0. Set $n = 0$

2. Define the scalar γ^n:

$$\begin{aligned}
\gamma^n &= 0 \quad (\text{if } n = 0) \\
\gamma^n &= \frac{\langle S'(\beta^n), S'(\beta^{n-1}) - S'(\beta^n)\rangle}{\|S'(\beta^{n-1})\|^2} \quad (\text{if } n > 0) \\
&= \frac{S'(\beta^n)^\top \left[S'(\beta^{n-1}) - S'(\beta^n)\right]}{S'(\beta^{n-1})^\top S'(\beta^{n-1})}
\end{aligned}$$

3. Define the direction p^n:

$$\begin{aligned}
p^0 &= -S'(\beta^0) \quad (\text{if } n = 0) \\
p^n &= -S'(\beta^n) + \gamma^n p^{n-1} \quad (\text{if } n > 0).
\end{aligned}$$

4. Define the stepsize α^n:

$$\begin{aligned}\alpha^n &= \frac{\langle S'(\beta^n), p^n \rangle}{\|D_{p^n} T(d, \cdot)\|^2} \\ &= \frac{(S'(\beta^n))^\top p^n}{(Xp^n)^\top (Xp^n)}.\end{aligned}$$

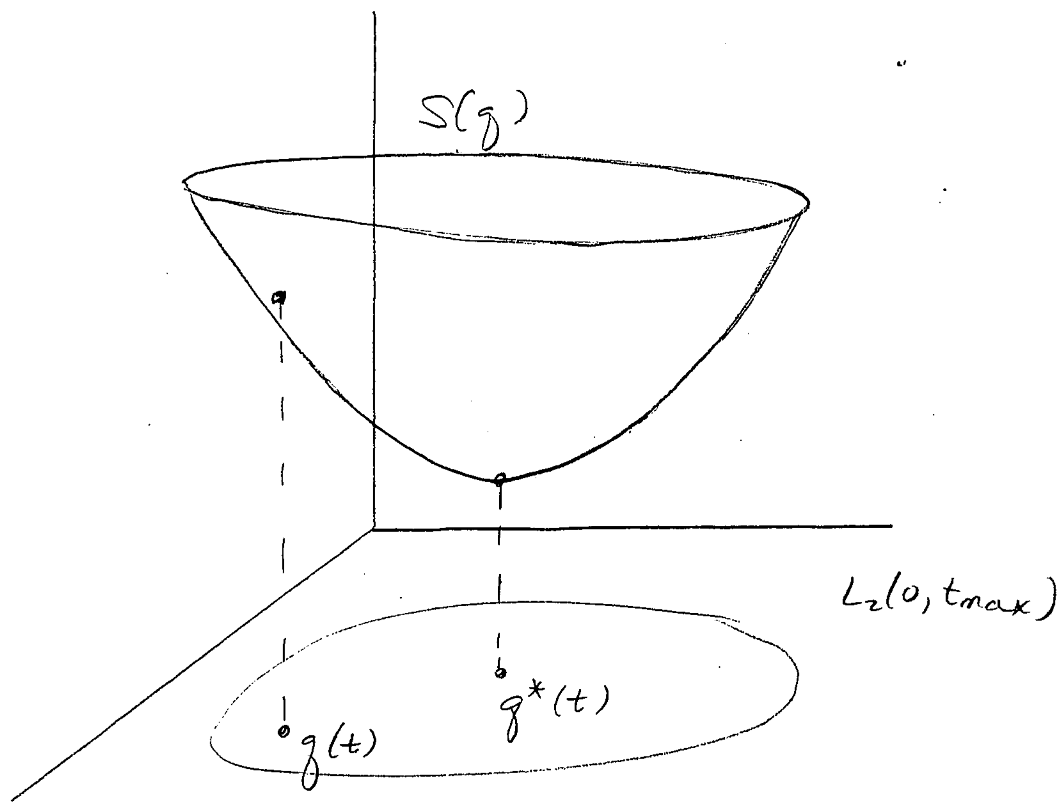
5. Set $\beta^{n+1} = \beta^n + \alpha^n p^n$.

6. If $\| \beta^{n+1} - \beta^n \|_{\mathbb{R}^p} < \epsilon$, stop. Otherwise, set $n = n + 1$ and go to step #2.

Lecture #3

Implementation of Descent Methods
for the
Infinite Dimensional Minimization Problem:

$$\min_{q \text{ in } L_2(0, t_{\max})} S(q)$$



Implementation of Descent Methods
for the
Infinite Dimensional Minimization Problem: Method of Steepest Descent for:

$$\min_{q \text{ in } L_2(0,t_{\max})} S(q).$$

Steps:

1. Pick an initial guess $q^0(t)$ in $L_2(0, t_{\max})$. For example, $q^0(t) = 0,\ 0 \le t \le t_{\max}$. Set $n = 0$.

2. Calculate $S'(q^n)$ and define the n^{th} "direction function" $p^n(t)$ (p^n is also in $L_2(0, t_{\max})$),
$$p^n = -S'(q^n).$$

3. Calculate α^n which satisfies the necessary condition for $S(q^n + \alpha p^n)$ to be minimized at α:
$$\frac{d}{d\alpha} S(q^n + \alpha p^n) = 0,$$
where q^n and p^n are given in previous steps. For the IHCP,
$$\alpha^n = \frac{\|p^n\|^2}{\|D_{p^n}T(d,\cdot)\|^2},$$
where $\|\cdot\|$ denotes the $L_2(0, t_{\max})$ norm. That is,
$$\alpha^n = \frac{\int_0^{t_{\max}} |p^n(t)|^2\, dt}{\int_0^{t_{\max}} |D_{p^n}T(d,t)|^2\, dt}.$$

4. Set
$$q^{n+1}(t) = q^n(t) + \alpha^n p^n(t), \quad 0 \le t \le t_{\max}.$$

5. If
$$\| q^{n+1} - q^n \|^2 \equiv \int_0^{t_{\max}} |q^{n+1}(t) - q^n(t)|^2\, dt < \epsilon^2,$$
stop. Otherwise, set $n = n + 1$ and go to step #2.

Implementation of Method of Steepest Descent to minimize $S(q)$ for IHCP:

$$\min_{q \text{ in } L_2(0,t_{\max})} S(q) = \min_{q \text{ in } L_2(0,t_{\max})} \int_0^{t_{\max}} |Y(t) - T(d,t;q)|^2 \, dt.$$

1. **Pick an initial guess,** $q^0(t)$ in $L_2(0, t_{\max})$. Set $n = 0$.

2. **Calculation of** $p^n(t)$: $(p^n(t) = -S'(q^n))$

   (a) First solve the original IHCP equations forward in time for the temperature $T(x,t;q^n)$:

$$\begin{aligned} \rho c \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\ T(x,0) &= 0 \\ kT_x|_{x=L} &= 0 \\ -kT_x|_{x=0} &= q^n(t) \quad \text{(use current } q^n) \end{aligned}$$

   (b) Compute the residual (measurement/model error) function corresponding to the current $q^n(t)$:

$$Y(t) - T(d,t;q^n), \quad 0 \le t \le t_{\max}.$$

   (c) Now solve the adjoint equations, backward in time, for $\psi(x,t;q^n)$:

$$\begin{aligned} \frac{\partial}{\partial t}(c\rho\psi) &= -\frac{\partial}{\partial x}(k\frac{\partial \psi}{\partial x}) \quad 0 < t < t_{\max}, \quad 0 < x < L \\ \psi(x, t_{\max}) &= 0 \\ k\psi_x|_{x=L} &= 0 \\ -k\psi_x|_{x=0} &= 0 \\ \psi(d^+,t) - \psi(d^-,t) &= 0 \\ -\left(k\frac{\partial \psi}{\partial x}(d^+,t) - k\frac{\partial \psi}{\partial x}(d^-,t)\right) &= Y(t) - T(d,t;q^n). \end{aligned}$$

   Note that the solution ψ at the current step is driven by the current residual $Y(t) - T(d,t;q^n)$, i.e., the error between measurements and model associated with the current iterate $q^n(t)$.

   Set

$$p^n(t) = -S'(q^n)(t) = -\psi(0,t;q^n), \quad 0 \le t \le t_{\max}.$$

   (Note that $p^n(t)$ is a function in $L_2(0, t_{\max})$.)

3. **Calculate α^n:** $(\alpha^n = \|p^n\|^2/\|D_{p^n}T(d,\cdot)\|^2)$.

   (a) First compute
   $$\| p^n \|^2 = \int_0^{t_{\max}} |\psi(0,t;q^n)|^2 \, dt.$$

   (b) Solve the sensitivity equations for $\theta(x,t;p^n)$ (forward in time):
   $$\begin{aligned} \rho c \frac{\partial \theta}{\partial t} &= \frac{\partial}{\partial x}\left(k \frac{\partial \theta}{\partial x}\right), \quad 0 < t < t_{\max}, \quad 0 < x < L \\ \theta(x,0) &= 0 \\ k\theta_x|_{x=L} &= 0 \\ -k\theta_x|_{x=0} &= p^n(t) \\ &= -\psi(0,t;q^n) \quad \text{(adjoint variable)} \end{aligned}$$

   (c) Then, $D_{p^n}T(d,t) = \theta(d,t;p^n)$ and
   $$\|D_{p^n}T(d,\cdot)\|^2 = \int_0^{t_{\max}} |\theta(d,t;p^n)|^2 \, dt.$$

   Set $\alpha^n = \|p^n\|^2/\|D_{p^n}T(d,\cdot)\|^2$.

4. The next (functional) q-iterate is given by
   $$q^{n+1}(t) = q^n(t) + \alpha^n p^n(t), \quad 0 \le t \le t_{\max}.$$

5. If $\| q^{n+1} - q^n \|^2 \equiv \int_0^{t_{\max}} |q^{n+1}(t) - q^n(t)|^2 \, dt < \epsilon^2$, stop. Otherwise, set $n = n+1$ and go to step #2.

Note: For each updated $q^{n+1}(t)$, we must first solve *three* partial differential equations. Because of the forward-backward-forward nature of the three sets of equations, they *cannot* be solved simultaneously.

Question: In *implementing* these "infinite dimensional schemes" on a computer, don't we have to *discretize*? That is, doesn't the problem again become *finite dimensional*?

Making the assumption that the actual "real world" problem is the infinite dimensional one, namely,

$$\min_{q \text{ in } L_2(0,t_{\max})} S(q) = \min_{q \text{ in } L_2(0,t_{\max})} \left\{ \frac{1}{2} \int_0^{t_{\max}} |Y(t) - T(d,t;q)|^2 \, dt \right\},$$

there are two approaches that we have discussed:

- Discretize first, optimize later (usual approach)
- Optimize first, discretize later ("adjoint equation" approach)

Are the two approaches equivalent?

Discretize first, optimize later

- Discretize q via an *a priori* parameterization.
- Minimization problem is

$$\min_{\beta \text{ in } \mathbb{R}^p} S(\beta).$$

- Pick Descent Method;
  i.e., make initial guess β^0 and determine formulas for α^n and p^n in the iteration scheme
  $\beta^{n+1} = \beta^n + \alpha^n p^n$.
  These formulas require $S'(\beta^n)$.
- To compute $S'(\beta^n)$ for each n:
  - Solve (via approximation technique, e.g. finite difference method) the IHCP equations forward in time for *approximate* $T^M(x,t;\beta^n)$.
  - Multiply $(-X^\top)$ times the residual $\left[Y - T^M(d,\cdot;\beta^n)\right]$.
- Update $\beta^{n+1} = \beta^n + \alpha^n p^n$, and continue.

Optimize first, discretize later

- Minimization problem is

$$\min_{q \text{ in } L_2(0,t_{\max})} S(q).$$

- Pick Descent Method; i.e., make initial guess $q^0(t)$ and determine formulas for α^n and p^n in the iteration
  $q^{n+1}(t) = q^n(t) + \alpha^n p^n(t)$.
  These formulas require $S'(q^n)$.
- To compute $S'(q^n)$ for each n:
  - Solve (via approximation technique, e.g. finite difference method) the IHCP equations forward in time for *approximate* $T^M(x,t;q^n)$.
  - Solve (via approximation technique) the adjoint equations backward in time for *approximate* $\psi^M(x,t;q^n)$.

  Then, $S'(q^n) \approx \psi^M(0,t;q^n)$
  (since $\psi^M(x,t;q^n) \approx \psi(x,t;q^n)$)
- Update $q^{n+1}(t) = q^n(t) + \alpha^n p^n(t)$, where
  $p^n(t) = -\psi^M(0,t;q^n) \approx -S'(q^n)(t)$
  is used. At this step, the parameter q^{n+1} is *discretized* (e.g., M-dimensional).

Clearly, the two approaches are *not* equivalent. Hopefully, however, as $p \to \infty$ and $M \to \infty$, the same optimal functional parameter, $q^* = q^*(t)$, is reached.

Regularizing Fit-to-Data Criteria

Suppose that a regularizing term is added to $S(q)$, e.g., for given (fixed) $\delta > 0$, we have the **zero$^{\text{th}}$ order regularizing criterion**

$$S_\delta(q) = \frac{1}{2}\int_0^{t_{\max}} |Y(t) - T(d,t;q)|^2\,dt \;+\; \frac{\delta}{2}\int_0^{t_{\max}} |q(t)|^2\,dt,$$

or, the **first order regularizing criterion**

$$S_\delta(q) = \frac{1}{2}\int_0^{t_{\max}} |Y(t) - T(d,t;q)|^2\,dt \;+\; \frac{\delta}{2}\int_0^{t_{\max}} \left[|q(t)|^2 + |q'(t)|^2\right]\,dt.$$

Then we may still implement infinite dimensional descent methods for minimizing $S_\delta(q)$.

The zero$^{\text{th}}$ order regularization is straightforward:

- Pick $q^0(t)$ and define the iteration $q^{n+1}(t) = q^n(t) + \alpha^n p^n(t)$, where the formulas for α^n and p^n are analogous to those given before, requiring now $S'_\delta(q^n)$.
- Computation of $S'_\delta(q)$ for any q:
  - Compute $S_\delta(q+\Delta q) - S_\delta(q)$ (where q and Δq are given functions in $L_2(0,t_{\max})$) and drop terms nonlinear in Δq. What remains is the *directional* derivative $D_{\Delta q}S_\delta(q)$.
  - Where, for the *non*-regularized problem, we had

$$\begin{aligned} D_{\Delta q}S(q) &= \underbrace{\int_0^{t_{\max}}(Y(t) - T(d,t;q))}_{\text{residual}}\underbrace{(-D_{\Delta q}T(d,t))}_{\substack{\text{directional}\\\text{derivative}\\\text{of T at } q}}\,dt \\ &= \int_0^{t_{\max}} \underbrace{\psi(0,t;q)}_{S'(q)}\cdot\Delta q(t)\,dt, \end{aligned}$$

(ψ the solution of the adjoint equations), we now have

$$\begin{aligned} D_{\Delta q}S_\delta(q) &= \underbrace{\int_0^{t_{\max}}(Y(t) - T(d,t;q))}_{\text{residual}}\underbrace{(-D_{\Delta q}T(d,t))}_{\substack{\text{directional}\\\text{derivative}\\\text{of T at } q}}\,dt \\ &\quad + \;\delta\underbrace{\int_0^{t_{\max}} q(t)\cdot\Delta q(t)}_{\text{reg. term}}\,dt \\ &= \int_0^{t_{\max}} \underbrace{[\psi(0,t;q) + \delta q(t)]}_{S'_\delta(q)}\cdot\Delta q(t)\,dt. \end{aligned}$$

It follows that

$$S'_\delta(q)(t) = \psi(0,t;q) + \delta q(t).$$

- Thus, for the $(n+1)^{\text{st}}$ iterate, $q^{n+1}(t) = q^n(t) + \alpha^n p^n(t)$, we perform the following:
  - Solve the IHCP equations forward in time and compute the residual

$$Y(t) - T(d,t;q^n).$$

  - Solve the *usual* adjoint equations (with jump condition given by $Y(t)-T(d,t;q^n)$) backward in time for $\psi(x,t;q^n)$.

Then

$$p^n(t) = -S'_\delta(q^n)(t) = -[\psi(0,t;q^n) + \delta q^n(t)].$$

For the first order regularizing criterion,

$$S_\delta(q) = \frac{1}{2}\int_0^{t_{\max}} |Y(t) - T(d,t;q)|^2\, dt + \frac{\delta}{2}\int_0^{t_{\max}} \left[|q(t)|^2 + |q'(t)|^2\right]\, dt,$$

we are actually minimizing $S_\delta(q)$ over a *new parameter space.* Where before q belonged to $L_2(0,t_{\max})$ (the square-integrable functions), now q must have a square-integrable first derivative; i.e., we take q in $W^{1,2}(0,t_{\max})$ (the square-integrable functions which have square-integrable derivatives).

Additionally, where before for $L_2(0,t_{\max})$ we had the scalar product

$$\langle q, \tilde{q}\rangle \equiv \int_0^{t_{\max}} q(t)\tilde{q}(t)\, dt,$$

we must define for $W^{1,2}(0,t_{\max})$ a new scalar product which takes into account the *new derivative information* about q. In this case we define the scalar product by

$$\langle q, \tilde{q}\rangle \equiv \int_0^{t_{\max}} [q(t)\tilde{q}(t) + q'(t)\tilde{q}'(t)]\, dt.$$

This change in scalar product leads to a new complications in defining $S'_\delta(q)$.

Before, we rewrote the directional derivative as

$$\begin{aligned} D_{\Delta q} S_\delta(q) &= \langle \text{“function of } t\text{”}, \Delta q \rangle \\ &= \int_0^{t_{\max}} (\text{“function of t”}) \cdot \Delta q(t)\, dt. \end{aligned}$$

and identified (via definition of the adjoint ψ)

$$S'_\delta(q) = \text{“function of t”} = \psi(0, t; q) + \delta q(t).$$

Now we must rewrite the directional derivative as

$$\begin{aligned} D_{\Delta q} S_\delta(q) &= \langle \text{“function of } t, \Delta q \rangle \\ &= \int_0^{t_{\max}} [\, (\text{“function of t”}) \cdot \Delta q(t) + (\text{“\textit{same} function of t”})' \cdot (\Delta q)'(t) \,]\, dt \end{aligned}$$

and, once this is done, identify

$$S'_\delta(q) = \text{“function of t”}.$$

We may show, for the first order regularization criterion, that

$$\begin{aligned} D_{\Delta q} S_\delta(q) &= \underbrace{\int_0^{t_{\max}} (Y(t) - T(d,t;q))}_{\text{residual}} \underbrace{(-D_{\Delta q} T(d,t))}_{\substack{\text{directional}\\ \text{derivative}\\ \text{of T at } q}} dt + \delta \underbrace{\int_0^{t_{\max}} [q(t) \cdot \Delta q(t) + q'(t) \cdot (\Delta q)'(t)]\, dt}_{\text{reg. term}} \\ &= \int_0^{t_{\max}} \psi(0,t;q) \cdot \Delta q(t)\, dt + \int_0^{t_{\max}} [\delta q(t) \cdot \Delta q(t) + \delta q'(t) \cdot \Delta q'(t)]\, dt. \end{aligned}$$

Suppose we could find some function $\rho = \rho(t)$ in $W^{1,2}(0, t_{\max})$ such that, from the last equality,

$$\int_0^{t_{\max}} \psi(0,t;q) \cdot \Delta q(t)\, dt = \int_0^{t_{\max}} [\rho(t) \cdot \Delta q(t) + \rho'(t) \cdot (\Delta q)'(t)]\, dt.$$

We could then identify $S'_\delta(q)$ with $\rho + \delta q$, a function of t.

To construct such a ρ, we first note that we may integrate by parts in the last equality above and obtain

$$\begin{aligned} \int_0^{t_{\max}} [\rho(t)\Delta q(t) + \rho'(t) \cdot (\Delta q)'(t)]\, dt &= \int_0^{t_{\max}} \rho(t) \cdot \Delta q(t)\, dt \\ &\quad + \rho'(t_{\max}) \cdot \Delta q(t_{\max}) - \rho'(0) \cdot \Delta q(0) - \int_0^{t_{\max}} \rho''(t) \cdot \Delta q(t)\, dt \\ &= \rho'(t_{\max}) \cdot \Delta q(t_{\max}) - \rho'(0) \cdot \Delta q(0) + \int_0^{t_{\max}} [-\rho''(t) + \rho(t)] \cdot \Delta q(t) \end{aligned}$$

Therefore, we select ρ such that it satisfies the following two-point boundary value problem,

$$\begin{aligned} -\ \rho''(t) + \rho(t) &= \psi(0,t;q), \quad 0 < t < t_{\max}, \\ \rho'(0) = \rho'(t_{\max}) &= 0, \end{aligned}$$

and thus find that

$$\begin{aligned} D_{\Delta q} S_\delta(q) &\equiv \int_0^{t_{\max}} [(\psi(0,t;q) + \delta q(t))\,\Delta q(t) + \delta q'(t)(\Delta q)'(t)]\; dt \\ &= \int_0^{t_{\max}} \{[\rho(t) + \delta q(t)]\cdot \Delta q(t) + [\rho(t) + \delta q(t)]'(t)\cdot(\Delta q)'(t)\}\; dt \end{aligned}$$

as needed. It follows that

$$S_\delta'(q) = \rho + \delta q,$$

a function of t.

Thus, for the first order regularization criterion, we add a *fourth* equation that must be solved at each iterate of our parameter value, namely the two-point boundary value problem for ρ. The equations in ρ are solved *after* the adjoint equations are solved.

Final Comment on First Order Regularization:

Because the scalar product changes, the norm $\|\cdot\|$ used in the (Steepest Descent) formula for α^n also changes. Now we must compute

$$\|p^n\|^2 = \int_0^{t_{\max}} [\,|p^n(t)|^2 + |(p^n)'(t)|^2\,]\, dt.$$