10.34, Numerical Methods Applied to Chemical Engineering
Professor William H. Green
Lecture #18: Optimization. Sensitivity Analysis. Introduction: Boundary Value Problems (BVPs).

Summary: Optimization with Constraints

$\min_{\underline{x}} f(\underline{x})$ such that $c_m(\underline{x}) - s_m = 0$

$s_m \geq 0 \quad m = 1 \ldots N_{\text{inequalities}}$

$\min_{\underline{x}} f(\underline{x}) + \xi(c-s)^2$ $\quad s_m = 0 \quad m > N_{\text{inequalities}}$

penalty method, second term $\xi(c-s)^2$ is optional

KKT conditions: at constrained (local) minimum:

Augmented Lagrangian (LA) {see book}

$\underline{\nabla}_x f - \Sigma_m(\lambda_m \underline{\nabla}_x c_m) = \underline{0} \rightarrow$

$c_m - s_m = 0$

$\lambda_m c_m = 0$

$s_m \geq 0 \qquad m = 1 \ldots N_{\text{inequalities}}$

$s_m = 0 \qquad$ equalities

$$\underline{x} = \begin{pmatrix} \underline{c} \\ \underline{\lambda} \\ \underline{s} \end{pmatrix} \qquad \underline{F}(\underline{x}) = \underline{0} = \begin{pmatrix} \nabla f - \sum \lambda_m \nabla c_m \\ c_m - s_m \\ \lambda_m c_m \\ s_m \end{pmatrix}$$

Newton → SQP

If everything is linear: → SIMPLEX (i.e. many business problems)

$g(\underline{x}) = 0 \rightarrow \underline{x}_N = G(x_1, \ldots, x_{N-1})$

Unconstrained → trust region Newton-type BFGS

gigantic → conjugate gradient

In Chemical Engineering, the problems often involve models with differential equations:

$$f(\underline{x}) = \sum_i w_i \Big(\underbrace{\overset{\text{cost}}{Y_i(t_o;\underline{x})}}_{\text{what we need}} - \underbrace{\overset{\text{return}}{Y_i(t_f;\underline{x})}}_{\text{what we produce}} \Big)$$

knobs
(can adjust)
feed composition

Need Jacobian of G with respect to Y; need in stiff solver to solve.

$$\frac{d\underline{Y}}{dt} = G(\underline{Y};\underline{x}) \qquad \underline{Y}(t_0) = \underline{Y}_0(\underline{x})$$

Need gradient and f.

To use all of our methods, we need to be able to compute: $\frac{\partial f}{\partial x_j} = \sum_i w_i \left(\frac{\partial Y_{\sigma,i}}{\partial x_j} - \frac{\partial Y_i(t_f)}{\partial x_j} \right)$

how do you compute this?

$$\frac{\partial}{\partial x_j}\left(\frac{\partial Y_i}{\partial t}\right) = \underbrace{\left(\sum_n \frac{\partial G}{\partial Y_n}\frac{\partial Y_n}{\partial x_j}\right)}_{\text{chain rule}} + \frac{\partial G_i}{\partial x_j}$$

{"sensitivity of $Y_i(t_f)$ to $\underline{x}_j$"}

$$\frac{\partial}{\partial t}\left(\frac{\partial Y_i}{\partial x_j}\right)$$

s_{ij}

$$\boxed{\frac{d}{dt} s_{ij} = \left(\sum_n \frac{\partial G_i}{\partial Y_n} s_{nj}\right) + \frac{\partial G_i}{\partial x_j}} \rightarrow \underline{\nabla} f$$

(for every $\underline{x}$ we get an $\underline{\nabla}$f that can be used for optimization)

solve this with initial conditions

$\underline{\underline{J}}$ {Jacobian of $\underline{G}$}

Have n^2 differential equations; stiff; linear in s.

Sensitivity Analysis

Programs to do this: DASPK
DAEPACK
DSL485
DASAC

SOLVE for s and f simultaneously

$$\frac{d}{dt} s_{ij} = \left(\sum_n \frac{\partial G_i}{\partial Y_n} s_{nj}\right) + \frac{\partial G_i}{\partial x_j}$$

Initial Conditions

What is $s_{ji}(t_0)$?

$s_{ji}(t_0) = 0$ {most knobs}

$s_{ji}(t_0) = 1$ {for adjustment of $\underline{Y}_0$}

Professor Barton teaches an advanced course in optimization.

Boundary Value Problems (BVPs)

Conservation Laws: $\partial\phi/\partial t = \underbrace{-\underline{\nabla}\cdot(\phi\underline{v})}_{\text{convection}} \underbrace{- \underline{\nabla}\cdot\underline{\underline{J}}_D}_{\text{diffusion}} + \underbrace{S(\phi)}_{\text{reaction}}$

$\underline{\underline{J}}_D = \underline{\underline{\Gamma}} + \underline{\nabla}\phi$

isotropic: $\underline{\underline{J}}_D = -c\underline{\nabla}\phi$

for steady-state, isotropic: $\boxed{0 = -\underline{\nabla}\cdot(\phi\underline{v}) - c\underline{\nabla}^2\phi + S(\phi)}$ $\forall\ \underline{x}$

Laplacian

Boundary conditions:

Dirichlet ϕ(boundary) = number

von Neumann $\nabla\phi$(boundary) = number or 0

Symmetry $\dfrac{\partial\phi}{\partial x_j} = 0$

$\phi(\underline{x})$ infinite {rare to find exact}

$\phi_{\text{approx}}(\underline{x}) = f(\underline{x};\ \underline{c})$ adjust: large finite number (10^4)

Basis function expansions $\phi_{\text{approx}} = \sum_{n=1}^{N_{basis}} c_n \Psi_n(\underline{x})$

$$\int_{\substack{m=1,N_{basis}\\ B.C.}} \Psi_m(\underline{x}) \underbrace{\left(-\nabla\cdot(\phi_{approx}\,\underline{v}) + c\nabla^2\phi_{approx} + s(\phi_{approx})\right)}_{\Re(\underline{x},\phi_{approx},\underline{c})} = 0$$

called "Residual"

$\phi_{approx}^{(x)} = f(\underline{x};\{\phi_i\})$ $\phi_{approx}(\underline{x_i})$

$\{\underline{x_i}\} = \text{mesh grid}$

some interpolation

Finite difference approximation to differential equation

$$\left.\frac{\partial\phi}{\partial x}\right|_{\underline{x_i}} \approx \frac{\phi_{i+1}-\phi_{i-1}}{x_{i+1}-x_{i-1}}$$

State how you did approximation because there are many ways to do it

Dirichlet ϕ(boundary) $\{\phi\}$ i = 1,N

$$\left.\frac{\partial\phi}{\partial x}\right|_{x_1} = \frac{\phi_2-\phi_0}{x_2-x_0} \leftarrow \phi_{B.C.}$$

von Neumann

$\left.\frac{\partial \phi}{\partial x}\right|_{x_0}$ given ϕ_0? *Usual* → 2^{nd} order polynomials

$$\phi(\underline{x}) = \phi(x_0) + \left.\frac{\partial \phi}{\partial x}\right|_{x_0}(x - x_0) + \frac{1}{2}\left.\frac{\partial^2 \phi}{\partial x^2}\right|_{x_0}(x - x_0)^2$$

unknown known unknown

✓ ✓

$$\phi_0 = f(\phi_1, \phi_2) \qquad \left.\frac{\partial \phi}{\partial x}\right|_{x_1} = \frac{\phi_2 - f(\phi_1, \phi_2)}{x_2 - x_0}$$

$$\phi(x_1) = \phi_0 + \left.\frac{\partial \phi}{\partial x}\right|_{x_0}(x_1 - x_0) + \frac{1}{2}\left.\frac{\partial^2 \phi}{\partial x^2}\right|_{x_0}(x_1 - x_0)^2 \ldots$$

$$\phi(x_2) = \phi_0 + \left.\frac{\partial \phi}{\partial x}\right|_{x_0}(x_2 - x_0) + \frac{1}{2}\left.\frac{\partial^2 \phi}{\partial x^2}\right|_{x_0}(x_2 - x_0)^2 \ldots$$

$$\phi_0 = \frac{\phi_1 \frac{(x_2 - x_0)^2}{(x_1 - x_0)^2} - \phi_2}{\frac{(x_2 - x_0)^2}{(x_1 - x_0)^2} - 1} \qquad \text{for } \left.\frac{\partial \phi}{\partial x}\right|_{x_0} = 0$$

If Δx uniform, $\phi_0 = \frac{4\phi_1 - \phi_2}{3}$

This is how you find out B.C. with second order polynomial schemes and a finite difference approximation.