CH 3 NEWTON-TYPE OPTIMIZATION

NLP WITH INEQUALITIES

NONLINEAR INT. POINT (IP) (BARRIER)

SEQUENTIAL QUADRATIC PROGRAMMING (SQP)

$$\min_x \; f(x) \quad \text{s.t.} \quad g(x) = 0, \; h(x) \le 0$$

fmincon

QP

x_k, B_k →

$$\min_x \; \nabla f(x_k)^T (x - x_k) + \frac{1}{2}(x - x_k)^T B_k (x - x_k)$$

$$\text{s.t.} \quad g(x_k) + \nabla g(x_k)^T (x - x_k) = 0$$

$$h(x_k) + \nabla h(x_k)^T (x - x_k) \le 0$$

→ x_{k+1}, λ_{k+1}, μ_{k+1}

$B_k \succcurlyeq 0 \Rightarrow$ QP IS CONVEX

3.4 * GLOBALISATION STRATEGIES

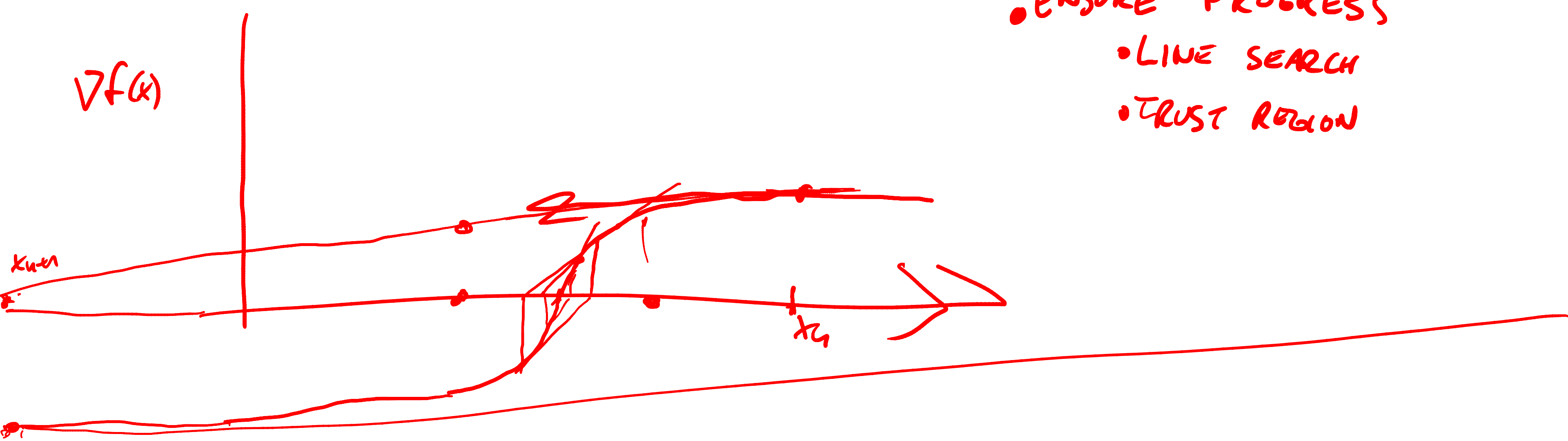
WOULD LIKE

"$f(x_{n+1}) \leq f(x_n)$"

NEED:

- MEASURE OF PROGRESS FOR CONSTRAINED OPT. ("MERIT FUNCTION")

- ENSURE PROGRESS
  - LINE SEARCH
  - TRUST REGION



CH 4: CALCULATING DERIVATIVES

HOW TO GET DERIVATIVES OF $F: \mathbb{R}^n \to \mathbb{R}^{n_F}$?

- BY HAND (BRAIN) : ERROR PRONE

$1{,}00\cdots01 - 1 = 0{,}0\cdots01$

- FINITE DIFFERENCES $\frac{\partial F(x)}{\partial x} \cdot p \approx \frac{F(x+\varepsilon \cdot p) - F(x)}{\varepsilon}$

ε TOO SMALL : NUMERICAL CANCELLATION ERRORS

ε TOO BIG : TAYLOR TRUNCATION ERRORS

$\varepsilon \sim 10^{-8}$ (FOR DOUBLE ACCURACY) 10^{-16}

- USE SYMBOLIC DIFFERENTIATION (MAPLE, MATHEMATICA, ...)
ACCURATE, LONG CODES, NO BOUND ON COST OF EVALUATION

- ALGORITHMIC ~~OPTIMIZATION~~ DIFFERENTIATION (AD)

4.2 Forward AD

EXAMPLE: $F(x) = \sin(x_1 x_2) + \exp(x_1 \cdot x_2 \cdot x_3)$ $\quad x \in \mathbb{R}^3, \; F \in \mathbb{R}$

$x_4 = x_1 \cdot x_2$

$x_5 = \sin(x_4)$

$x_6 = x_4 \cdot x_3$

$x_7 = \exp(x_6)$

$x_8 = x_5 + x_7$

$F(x) := x_8$

ELEMENTARY OPS.

AD - FORWARD MODE

$\dot{x}_4 = x_1 \cdot \dot{x}_2 + \dot{x}_1 \cdot x_2$

$\dot{x}_5 = \cos(x_4) \cdot \dot{x}_4$

$\dot{x}_6 = x_4 \cdot \dot{x}_3 + \dot{x}_4 \cdot x_3$

$\dot{x}_7 = \underbrace{\exp(x_6)}_{= x_7} \cdot \dot{x}_6$

$\dot{x}_8 = \dot{x}_5 + \dot{x}_7$

$\dot{F} = \dot{x}_8$

$\| \; \frac{\partial F}{\partial x}(x) \cdot p$

$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} := \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$

"DOT-QUANTITIES"

$\frac{\partial F}{\partial x}(x) \cdot p = \frac{dF}{dt}(x + tp)\Big|_{t=0}$

$\mathrm{COST}\left(\frac{\partial F}{\partial x} \cdot p\right) \leq C \cdot \mathrm{COST}(F)$

$\uparrow$

$1-3$

SIMILAR COST AS FD, BUT HIGH ACCURACY (BOTH CPU TIME AND MEMORY)

TO GET FULL JACOBIAN:

$$\frac{\partial F}{\partial x}(x) = \left[\frac{\partial F}{\partial x}(x)\cdot \begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix} \;\middle|\; \frac{\partial F}{\partial x}(x)\begin{bmatrix}0\\1\\0\\\vdots\\0\end{bmatrix} \;\middle|\; \cdots \;\middle|\; \frac{\partial F}{\partial x}(x)\begin{bmatrix}0\\\vdots\\0\\1\end{bmatrix}\right]$$

p "SEED VECTORS"

4.8.1 "IMAGINARY TRICK"

REAL F $\longrightarrow$ COMPLEX F

$x_i \longrightarrow x_i + \varepsilon \cdot j \cdot \dot{x}_i$

$$\boxed{\frac{\mathrm{Im}\left(F(x+\varepsilon\cdot p\cdot j)\right)}{\varepsilon} = \frac{\partial F}{\partial x}(x)\cdot p \quad \text{WITH } \varepsilon = 10^{-100}}$$

(EQUIVALENT TO FORWARD-AD)

$$\tilde{x}_0 = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad \tilde{x}_1 = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ x_{n+1} \end{bmatrix}, \quad \ldots \quad \tilde{x}_m = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ \vdots \\ x_{n+m} \end{bmatrix} \qquad \Bigg\| \qquad \dot{\tilde{x}}_i = \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_{n+i} \end{bmatrix}$$

$$\tilde{x}_{i+1} = \tilde{\phi}_i(\tilde{x}_i) = \begin{bmatrix} x_1 \\ \vdots \\ x_{n+i} \\ \phi_i(\ldots) \end{bmatrix} \begin{matrix} \leftarrow \text{COPY} \\ \\ \leftarrow \text{ELEMENT. OPERATOR} \end{matrix}$$

$$F(x) = C \cdot \tilde{\phi}_{m-1}(\tilde{\phi}_{m-2}(\ldots))$$

$$C = [0 \ldots 0 \; 1]$$

$$\boxed{\frac{\partial F}{\partial x}(x) = J_F(x) = C \cdot \tilde{J}_{m-1} \cdot \tilde{J}_{m-2} \cdot \ldots \tilde{J}_0}$$

$$\tilde{J}_i = \frac{\partial \tilde{\phi}_i}{\partial \tilde{x}_i} = \begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & \ddots & \\ 0 & & & 1 \\ \hline * & & & * \end{bmatrix}$$

1-2 ENTRIES $\rightarrow$

$$\boxed{J_F \cdot p = C\Big(\tilde{J}_{m-1} \ldots \underbrace{\big(\tilde{J}_1 \cdot \underbrace{(\tilde{J}_0 \cdot p)}_{\text{VECTOR } = \dot{\tilde{x}}_1}\big)}_{\text{VECTOR } \dot{\tilde{x}}_2}\Big)}$$

$$\lambda^T J_F = ? \Rightarrow \text{BACKWARD AD}$$

$$\lambda \in \mathbb{R}^{n_F}$$

$$\lambda^T J_F = \left(\left(\lambda^T \cdot C \right) \cdot \tilde{J}_{m-1} \right) \cdot \tilde{J}_{m-2} \cdot \quad \cdots \quad \tilde{J}_0$$

$$J_F^T \cdot \lambda = \tilde{J}_0^T \left(\tilde{J}_1^T \left(\quad \cdots \quad \left(\underbrace{\tilde{J}_{m-1}^T \left(\underbrace{C^T \cdot \lambda}_{\bar{\tilde{x}}_m} \right)}_{\bar{\tilde{x}}_{m-1}} \cdots \right) \right) \right)$$

$$\boxed{\tilde{J}_i^T = \begin{pmatrix} 1 & & & * \\ & \ddots & & 0 \\ & & 1 & * \end{pmatrix}}$$

$$\boxed{\bar{x}_q = \bar{x}_q + * \cdot \bar{x}_{q'}}$$

$$\boxed{\cancel{\bar{x}_q = \bar{x}_q}}$$

x_i $\quad \bar{x}_i \quad$ $\begin{bmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_{n+m} \end{bmatrix}$ BAR QUANTITIES $\quad \bar{\tilde{x}}_m$

FOR $n_F = 1$, $C = (0 \dots 0\ 1)$ $(\lambda = 1)$

$$\lambda^T \frac{\partial F}{\partial x}(x) = \nabla F(x)^T$$

EXAMPLE:

$x_4 = x_1 x_2$

$x_5 = \sin(x_4)$

$x_6 = x_4 \cdot x_3$

$x_7 = \exp(x_6)$

$x_8 = x_5 + x_7$

$$C = (0 \dots 0\ \underset{\bar{x}_8}{\underset{\uparrow}{1}})$$

BACKWARD-AD:

$\bar{x}_1, \dots, \bar{x}_7 = 0$, $\bar{x}_8 = 1$

// $x_8 = x_5 + x_7$

$\bar{x}_5 = \bar{x}_5 + 1 \cdot \bar{x}_8$

$\bar{x}_7 = \bar{x}_7 + 1 \cdot \bar{x}_8$

// $x_7 = \exp(x_6)$

$\bar{x}_6 = \bar{x}_6 + \underline{\exp(x_6)} \cdot \bar{x}_7$

// $x_6 = x_4 \cdot x_3$

$\bar{x}_4 = \bar{x}_4 + \underline{x_3} \cdot \bar{x}_6$

$\bar{x}_3 = \bar{x}_3 + \underline{x_4} \cdot \bar{x}_6$

// $x_5 = \sin(x_4)$

$\bar{x}_4 = \bar{x}_4 + \underline{\cos(x_4)} \cdot \bar{x}_5$

// $x_4 = x_1 \cdot x_2$

$\bar{x}_1 = \bar{x}_1 + x_2 \cdot \bar{x}_4$

$\bar{x}_2 = \bar{x}_2 + x_1 \cdot \bar{x}_4$

$$\begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \end{bmatrix} = \nabla F(x)$$

$$\boxed{\mathrm{COST}(\lambda^T J_F) \le C \cdot \mathrm{COST}(F)}$$

$$C \simeq 1-4$$

FOR $F: \mathbb{R}^n \to \mathbb{R}$, $J_F = \nabla F(x)^T$

$$\mathrm{COST}(\nabla F(x)) \le C \cdot \mathrm{COST}(F)$$ INDEPENDENT OF n!

"CHEAP GRADIENT" RESULT FROM BACKWARDS AD

	FD	FORWARD AD	BACKWARD AD
ACCURACY	LOW	HIGH	HIGH
CPU TIME	OK	OK	OK / EXCITING
MEMORY	OK	OK	HIGH

help $\int$ min con