Reduced SQP methods

$$(*)\begin{cases} \min f(x) \\ \text{s.t. } c(x)=0 \end{cases} \qquad \begin{matrix} f: \mathbb{R}^n \to \mathbb{R} \\ c: \mathbb{R}^n \to \mathbb{R}^m \end{matrix}$$

$$L(x,\lambda) = f(x) + \lambda^T c(x)$$

At local min of $(*)$

$$\nabla L(x_*, \lambda_*) = 0$$

$$v^T \nabla^2 L(x_*, \lambda_*) v \geq 0 \quad \forall v \neq 0 \text{ in } \mathcal{N}(c'(x_*))$$

If second order suff cd^t^ are satisfied we have >0.

<u>SQP</u>: solve a seq. of quad problems.

$$\min \nabla f(x_k)^T \Delta x + \frac{1}{2} \Delta x^T H_k \Delta x$$
$$\text{s.t. } c(x_k) + c'(x_k)\Delta x = 0$$

$$x_{k+1} = x_k + \Delta x$$
$$\lambda_{k+1} = \lambda_k + \Delta \lambda$$

usually replace $\nabla^2_{xx} L(x_k, \lambda_k)$ by some H_k easier to compute (BFGS etc...)

however H_k in BFGS is enforced to be pos def when $\nabla^2_{xx} L(x_k, \lambda_k)$ needs not be!

Find nullspace of $c'(x_k) \in \mathbb{R}^{m\times n}$, $m<n$

1) $c'(x_k) = (\underbrace{B}_{m} \mid \underbrace{N}_{n-m})$, B = invertible

$$Z_k = \begin{pmatrix} -B_k^{-1} N \\ I \end{pmatrix} \in \mathbb{R}^{n\times(n-m)} \text{ of } rk\,(n-m)$$

Check $c'(x_k) Z_k = 0$

2) $c'(x_k)^T = Q_k \begin{pmatrix} \overbrace{R_k}^{m} \\ 0 \end{pmatrix} \begin{matrix} \}m \\ \}n-m \end{matrix}$, Q_k orthogonal $R_k = (\nabla)$

$$Q_k = (\underbrace{Y_k}_{m} \mid \underbrace{Z_k}_{n-m})$$

$$c'(x_k) Z_k = (R_k^T | 0) \begin{pmatrix} Y_k^T \\ Z_k^T \end{pmatrix} Z_k$$
$$= (R_k^T | 0) \begin{pmatrix} 0 \\ I \end{pmatrix} = 0$$

nice: columns of Z_k form an $\perp$ basis for $\mathcal{N}(c'(x_k))$

Let $Z_k \in \mathbb{R}^{n \times (n-m)}$ be a matrix w/ rank $(Z_k) = n-m$ and $c'(x_k) Z_k = 0$.

Let Δx_c be a particular sol of $c(x_k) + c'(x_k)\Delta x = 0$

Δx_c can be computed as follows:

1) $c'(x_k) = (B_k | N_k)$

$$\Delta x_c = \begin{bmatrix} -B_k^{-1} c(x_k) \\ 0 \end{bmatrix} \begin{matrix} \}m \\ \}n-m \end{matrix}$$

2) $c'(x_k)^T = Q_k R_k$

$c'(x_k) = R_k^T Y_k^T$

$$\underbrace{(R_k^T \;\; 0) \begin{pmatrix} Y_k^T \\ Z_k^T \end{pmatrix}}_{c'(x_k)} \Delta x + c(x_k) = 0$$

$$\text{Let } w = \begin{pmatrix} Y_k^T \\ Z_k^T \end{pmatrix} \Delta x = \begin{pmatrix} -(R_k^T)^{-1} c(x_k) \\ 0 \end{pmatrix} \begin{matrix} \}m \\ \}n-m \end{matrix}$$

$$\Delta x_c = (Y_k | Z_k)\, w = -Y_k (R_k^T)^{-1} c(x_k)$$

Now that we have Δx_c the sol of $c(x_k) + c'(x_k)\Delta x = 0$ is given by:

$$\Delta x = \Delta x_c + Z_k \, w^{\in \mathbb{R}^{n-m}}$$

constrained problem becomes unconstrained problem with $n-m$ unknowns:

$$\min_{w \in \mathbb{R}^{n-m}} \underbrace{\left(\nabla f(x_k)^T \Delta x_c + \frac{1}{2}\Delta x_c^T H_k \Delta x_c\right)}_{\text{const}} + \Big(\underbrace{Z_k^T(\nabla f(x_k) + H_k \Delta x_c)}_{\text{reduced grad}}\Big)^T w + \frac{1}{2} w^T \underbrace{Z_k^T H_k Z_k}_{\hat{H}_k = \text{reduced Hessian}} w$$

$\hat{H}_k$ is s.p.d iff H is spd. on $\mathcal{N}(c'(x^*))$

$\hat{H}_k$ perfect for BFGS update. $\leadsto$ reduced SQP method:

replace $Z_k^T \nabla_{xx} L(x_k, \lambda_k) Z_k$ by $\hat{H}_k$

instead of replacing $\nabla_{xx} L(x_k, \lambda_k)$ — by H_k

BFGS:

$$s_k = \underbrace{Z_k^T}_{\sim \text{ proj onto nullspace}} (x_{k+1} - x_k)$$

$$y_k = Z_k^T(\nabla_x L(x_{k+1}, \lambda_k) - \nabla L(x_k, \lambda_k))$$

$$\hat{H}_{k+1} = \hat{H}_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{\hat{H}_k s_k (\hat{H}_k s_k)^T}{s_k \hat{H}_k s_k}$$

We also need to approx $Z_k^T(\nabla f(x_k) + H_k \Delta x_c)$

1) $Z_k^T(\nabla f_k + H_k \Delta x_c) \sim Z_k^T \nabla f(x_c)$

especially if we are feasible $\Delta x_c = 0$.

2) $Z_k^T(\nabla f_k + \underbrace{\nabla_{xx} L(x_k, \lambda_k)}_{H_k} \Delta x_c)$

note: $Z_k^T \nabla f_k = Z_k^T \nabla_x L(x_k, \lambda_k)$

Thus we can use Taylor on $\nabla_x L(x_k, \lambda_k)$:

$$\nabla_x L(x_k + \Delta x_c, \lambda_k) \sim \nabla_x L(x_k, \lambda_k) + \nabla_{xx} L(x_k, \lambda_k) \Delta x_c$$

$\Rightarrow$ use approx:

$$Z_k^T \nabla_x L(x_k + \Delta x_c, \lambda_k).$$

(again approx is exact if we are feasible)

Reduced SQP (Local version)

Given x_0 $\hat{H}_0$

For $k = 0, 1, \ldots$

- compute nullspace rep. $Z_k \in \mathbb{R}^{n \times (n-m)}$
- compute particular sol Δx_c of linearized constraints
- $g_k = Z_k^T \nabla f(x_k)$ (approx of reduced gradient)
  or $Z_k^T \nabla L(x_k + \Delta x_c, \lambda_k)$
- solve $\hat{H}_k w = -g$
- set $x_{k+1} = x_k + \Delta x_c + Z_k w$
- update $\hat{H}_k$ to get $\hat{H}_{k+1}$.

For globalization: can still use l_1 merit function.

Questions: • how to approx λ_k?
• when to stop?

stop if $\|c(x)\| < \varepsilon$ and $\| Z_k^T \nabla f(x_k) \| < \varepsilon$.
or $\nabla L(x_k, \lambda_k)$

Why do we have $Z_k^T \nabla f(x_k) = 0 \Rightarrow \nabla L = 0$?

Because we can construct a λ s.t.

$$\nabla_x L(x,\lambda) = \nabla f(x) + c'(x)^T \lambda = 0$$

For example with method 1) for getting nullspace rep:

$$c'(x) = (B \mid N) \qquad Z_k = \begin{pmatrix} -B_k^{-1} N_k \\ I \end{pmatrix}$$

$$Z_k^T \nabla f_k = \begin{pmatrix} -B_k^{-1} N_k \\ I \end{pmatrix}^T \begin{pmatrix} \nabla f(x_k)|_B \\ \nabla f(x_k)|_N \end{pmatrix}$$

$$= N_k^T \big(\underbrace{-B_k^{-T} \nabla f(x_k)|_B}_{=\lambda} \big) + \nabla f(x_k)|_N$$

$$= 0$$

with such choice of λ we have:

$$\nabla_x L(x,\lambda) = \begin{bmatrix} \nabla f(x)|_B \\ \nabla f(x)|_N \end{bmatrix} + \underbrace{\begin{pmatrix} B^T \\ N^T \end{pmatrix}}_{c'(x)^T} \left[-B^{-T} \nabla f(x)|_B \right] = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Depending on choice of λ algorithm may not even converge to a point (x,λ) s.t. $\nabla L(x,\lambda) = 0$.

Reduced SQP methods are attractive if:

$$m \approx n \quad \Rightarrow \quad n - m \text{ small}$$

$\hat{H}_k$ small

Z_k easy to get.