Trust Region Method

Lectures for PHD course on Non-linear equations and numerical optimization

Enrico Bertolazzi

DIMS – Università di Trento

March 2005

Outline

- Newton and quasi-Newton methods search a solution iteratively by choosing at each step a search direction and minimize in this direction.
- An alternative approach is to to find a direction and a step-length, then if the step is successful in some sense the step is accepted. Otherwise another direction and step-length is chosen.
- The choice of the step-length and direction is algorithm dependent but a successful approach is the one based on trust region.

- Newton and quasi-Newton at each step (approximately) solve the minimization problem

$$\min\ m(\boldsymbol{x}_k + \boldsymbol{s}) = \mathsf{f}(\boldsymbol{x}_k) + \nabla \mathsf{f}(\boldsymbol{x}_k)\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T \boldsymbol{H}_k \boldsymbol{s}$$

in the case $\boldsymbol{H}_k$ is symmetric and positive definite (SPD).

- If $\boldsymbol{H}_k$ is SPD the minimum is

$$\boldsymbol{s} = -\boldsymbol{H}_k^{-1}\boldsymbol{g}_k, \qquad \boldsymbol{g}_k = \nabla \mathsf{f}(\boldsymbol{x}_k)^T$$

and $\boldsymbol{s}$ is the quasi-Newton step.

- If $\boldsymbol{H}_k = \nabla^2 \mathsf{f}(\boldsymbol{x}_k)$ and is SPD, then $\boldsymbol{s} = -\boldsymbol{H}_k^{-1}\boldsymbol{g}_k$ is the Newton step.

- If $\boldsymbol{H}_k$ is not positive definite, the search direction $-\boldsymbol{H}_k^{-1}\boldsymbol{g}_k$ may fail to be a descent direction and the previous minimization problem can have no solution.
- The problem is that the model $m(\boldsymbol{x}_k + \boldsymbol{s})$ is an approximation of $\mathsf{f}(\boldsymbol{x})$

$$m(\boldsymbol{x}_k + \boldsymbol{s}) \approx \mathsf{f}(\boldsymbol{x}_k + \boldsymbol{s})$$

  and this approximation is valid only in a small neighbors of $\boldsymbol{x}_k$.
- So that an alternative minimization problem is the following

$$\min \; m(\boldsymbol{x}_k + \boldsymbol{s}) = \mathsf{f}(\boldsymbol{x}_k) + \nabla \mathsf{f}(\boldsymbol{x}_k)\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\boldsymbol{H}_k\boldsymbol{s},$$

$$\text{Subject to } \; \|\boldsymbol{s}\| \leq \delta_k$$

  δ_k is the trust region of the model $m(\boldsymbol{x})$, i.e. the region where we trust the model is valid.

Algorithm (Generic trust region algorithm)

$\boldsymbol{x}$ *assigned;* δ *assigned;*

$\boldsymbol{g} \leftarrow \nabla \mathsf{f}(\boldsymbol{x})^T$; $\boldsymbol{H} \leftarrow \nabla^2 \mathsf{f}(\boldsymbol{x})^{-1}$;

while $\|\boldsymbol{g}\| > \epsilon$ **do**

  $\boldsymbol{s} \leftarrow \arg\min_{\|\boldsymbol{s}\| \leq \delta} m(\boldsymbol{x}+\boldsymbol{s}) = \mathsf{f}(\boldsymbol{x}) + \boldsymbol{g}^T\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\boldsymbol{H}\boldsymbol{s}$;

  pred $\leftarrow m(\boldsymbol{x}+\boldsymbol{s}) - m(\boldsymbol{x})$;

  ared $\leftarrow \mathsf{f}(\boldsymbol{x}+\boldsymbol{s}) - \mathsf{f}(\boldsymbol{x})$;

  if (*ared*/*pred*) $< \eta_1$ **then**

    $\boldsymbol{x} \leftarrow \boldsymbol{x}$; $\delta \leftarrow \delta\gamma_1$; — *reject step, reduce* δ

  else

    $\boldsymbol{x} \leftarrow \boldsymbol{x}+\boldsymbol{s}$; — *accept step, update* $\boldsymbol{H}$

    if (*ared*/*pred*) $> \eta_2$ **then**

      $\delta \leftarrow \max\{\delta, \gamma_2 \|\boldsymbol{s}\|\}$; — *enlarge* δ

    end if

  end if

end while

- The previous algorithm is based on two keys ingredients:
  1. The ratio $r = (\text{ared}/\text{pred})$ which is the ratio of the actual reduction and the predicted reduction.
  2. Enlarge or reduce the trust region δ.
- If the ratio r is between $0 < \eta_1 < r < \eta_2 < 1$ we have that the model is quite appropriate; we accept the step and do not modify the trust region.
- If the ratio r is small $r \leq \eta_1$ we have that the model is not appropriate; we do not accept the step and we must reduce the trust region by a factor $\gamma_1 < 1$
- If the ratio r is large $r \geq \eta_2$ we have that the model is very appropriate; we do accept the step and we enlarge the trust region factor $\gamma_2 > 1$
- The algorithm is quite insensitive to the constant η_1 and η_2. Typical values are $\eta_1 = 0.25$, $\eta_2 = 0.75$, $\gamma_1 = 0.5$ and $\gamma_2 = 3$.

Lemma

Let $\mathsf{f} : \mathbb{R}^n \mapsto \mathbb{R}$ *be twice continuously differentiable,* $\boldsymbol{H} \in \mathbb{R}^{n\times n}$ *symmetric and positive definite. Then the problem*

$$\min \; m(\boldsymbol{x}+\boldsymbol{s}) = \mathsf{f}(\boldsymbol{x}) + \nabla\mathsf{f}(\boldsymbol{x})\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\boldsymbol{H}\boldsymbol{s},$$

$$\text{Subject to } \|\boldsymbol{s}\| \leq \delta$$

is solved by

$$\boldsymbol{s}(\mu) \doteq -(\boldsymbol{H}+\mu\boldsymbol{I})^{-1}\boldsymbol{g}, \qquad \boldsymbol{g} = \nabla\mathsf{f}(\boldsymbol{x})^T$$

for the unique $\mu \geq 0$ *such that* $\|\boldsymbol{s}(\mu)\| = \delta$*, unless* $\|\boldsymbol{s}(0)\| \leq \delta$*, in which case* $\boldsymbol{s}(0)$ *is the solution. For any* $\mu \geq 0$*,* $\boldsymbol{s}(\mu)$ *defines a descent direction for* f *from* $\boldsymbol{x}$*.*

Proof. (1/2).

If $\|\boldsymbol{s}(0)\| \leq \delta$ then $\boldsymbol{s}(0)$ is the global minimum inside the trust region. Otherwise consider the Lagrangian

$$\mathcal{L}(\boldsymbol{s}, \mu) = a + \boldsymbol{g}^T \boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T \boldsymbol{H} \boldsymbol{s} + \frac{1}{2}\mu(\boldsymbol{s}^T \boldsymbol{s} - \delta^2),$$

where $a = \mathsf{f}(\boldsymbol{x})$ and $\boldsymbol{g} = \nabla \mathsf{f}(\boldsymbol{x})^T$. Then we have

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{s}}(\boldsymbol{s}, \mu) = \boldsymbol{H}\boldsymbol{s} + \mu\boldsymbol{s} + \boldsymbol{g} = 0 \quad \Rightarrow \quad \boldsymbol{s} = -(\boldsymbol{H} + \mu\boldsymbol{I})^{-1}\boldsymbol{g}$$

and $\boldsymbol{s}^T\boldsymbol{s} = \delta^2$. Remember that if $\boldsymbol{H}$ is SPD then $\boldsymbol{H} + \mu\boldsymbol{I}$ is SPD for all $\mu \geq 0$. Moreover the inverse of an SPD matrix is SPD. From

$$\boldsymbol{g}^T\boldsymbol{s} = -\boldsymbol{g}^T(\boldsymbol{H} + \mu\boldsymbol{I})^{-1}\boldsymbol{g} < 0 \qquad \text{for all } \mu \geq 0$$

follows that $\boldsymbol{s}(\mu)$ is a descent direction for all $\mu \geq 0$.

Proof. (2/2).

To prove the uniqueness consider expand the gradient $\boldsymbol{g}$ with the eigenvectors of $\boldsymbol{H}$

$$\boldsymbol{g} = \sum_{i=1}^{n} \alpha_i \boldsymbol{u}_i$$

$\boldsymbol{H}$ is SPD so that $\boldsymbol{u}_i$ can be chosen orthonormal. It follows

$$(\boldsymbol{H} + \mu \boldsymbol{I})^{-1}\boldsymbol{g} = (\boldsymbol{H} + \mu \boldsymbol{I})^{-1} \sum_{i=1}^{n} \alpha_i \boldsymbol{u}_i = \sum_{i=1}^{n} \frac{\alpha_i}{\lambda_i + \mu} \boldsymbol{u}_i$$

$$\left\| (\boldsymbol{H} + \mu \boldsymbol{I})^{-1}\boldsymbol{g} \right\|^2 = \sum_{i=1}^{n} \frac{\alpha_i^2}{(\lambda_i + \mu)^2}$$

and $\left\| (\boldsymbol{H} + \mu \boldsymbol{I})^{-1}\boldsymbol{g} \right\|$ is a monotonically decreasing function of μ. □

Remark

As a consequence of the previous Lemma we have:

- *as the ray of the trust region becomes smaller as the scalar* μ *becomes larger. This means that the search direction become more and more oriented toward the gradient direction.*
- *as the ray of the trust region becomes larger as the scalar* μ *becomes smaller. This means that the search direction become more and more oriented toward the Newton direction.*

Thus a trust region technique not only change the size of the step-length but also its direction. This results in a more robust numerical technique. The price to pay is that the solution of the minimization is more costly than the inexact line search.

Solving the constrained minimization problem

As for the line-search problem we have many alternative for solving the constrained minimization problem:

- We can solve accurately the constrained minimization problem. For example by an iterative method.
- We can approximate the solution of the constrained minimization problem.

as for the line search the accurate solution of the constrained minimization problem is not paying while a good cheap approximations is normally better performing.

Outline

The Newton approach (1/5)

- Consider the Lagrangian

$$\mathcal{L}(\boldsymbol{s},\mu) = a + \boldsymbol{g}^T\boldsymbol{s} + \frac{1}{2}\boldsymbol{s}^T\boldsymbol{H}\boldsymbol{s} + \frac{1}{2}\mu(\boldsymbol{s}^T\boldsymbol{s} - \delta^2),$$

where $a = \mathsf{f}(\boldsymbol{x})$ and $\boldsymbol{g} = \nabla\mathsf{f}(\boldsymbol{x})^T$.

- Then we can try to solve the nonlinear system

$$\frac{\partial\mathcal{L}}{\partial(\boldsymbol{s},\mu)}(\boldsymbol{s},\mu) = \begin{pmatrix} \boldsymbol{H}\boldsymbol{s} + \mu\boldsymbol{s} + \boldsymbol{g} \\ (\boldsymbol{s}^T\boldsymbol{s} - \delta^2)/2 \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix}$$

- Using Newton method we have

$$\begin{pmatrix} \boldsymbol{s}_{k+1} \\ \mu_{k+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{s}_k \\ \mu_k \end{pmatrix} - \begin{pmatrix} \boldsymbol{H} + \mu\boldsymbol{I} & \boldsymbol{s} \\ \boldsymbol{s}^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{H}\boldsymbol{s}_k + \mu_k\boldsymbol{s}_k + \boldsymbol{g} \\ (\boldsymbol{s}_k^T\boldsymbol{s}_k - \delta^2)/2 \end{pmatrix}$$

The Newton approach (2/5)

- A better approach is given by solving $\Phi(\mu) = 0$ where

$$\Phi(\mu) = \|\boldsymbol{s}(\mu)\| - \delta, \qquad \text{and} \qquad \boldsymbol{s}(\mu) = -(\boldsymbol{H} + \mu\boldsymbol{I})^{-1}\boldsymbol{g}$$

- To build Newton method we need to evaluate

$$\Phi(\mu)' = \frac{\boldsymbol{s}(\mu)^T\boldsymbol{s}(\mu)'}{\|\boldsymbol{s}(\mu)\|}, \qquad \boldsymbol{s}(\mu)' = (\boldsymbol{H} + \mu\boldsymbol{I})^{-2}\boldsymbol{g}$$

where to evaluate $\boldsymbol{s}(\mu)'$ we differentiate the relation

$$\boldsymbol{H}\boldsymbol{s}(\mu) + \mu\boldsymbol{s}(\mu) = \boldsymbol{g} \quad \Rightarrow \quad \boldsymbol{H}\boldsymbol{s}(\mu)' + \mu\boldsymbol{s}(\mu)' + \boldsymbol{s}(\mu) = \boldsymbol{0}$$

- Putting all in a Newton step we obtain

$$\mu_{k+1} = \mu_k - \frac{\|\boldsymbol{s}(\mu_k)\|}{\boldsymbol{s}(\mu_k)^T\boldsymbol{s}(\mu_k)'}(\|\boldsymbol{s}(\mu_k)\| - \delta)$$

The Newton approach (3/5)

- Newton step can be reorganized as follows

$$\begin{aligned} \boldsymbol{s}_k &= -(\boldsymbol{H} + \mu \boldsymbol{I})^{-1} \boldsymbol{g} \\ \boldsymbol{s}'_k &= -(\boldsymbol{H} + \mu \boldsymbol{I})^{-1} \boldsymbol{s}_k \\ \beta &= \sqrt{\boldsymbol{s}_k^T \boldsymbol{s}_k} \\ \mu_{k+1} &= \mu_k - \frac{\beta(\beta - \delta)}{\boldsymbol{s}_k^T \boldsymbol{s}'_k} \end{aligned}$$

- Thus Newton step require two linear system solution per step. However the coefficient matrix is the same so that only one LU factorization, thus the cost per step is essentially due to the LU factorization.

The Newton approach (4/5)

- Evaluating $\Phi(\mu)''$ we have

$$\Phi(\mu)'' = \frac{\|\boldsymbol{s}(\mu)\|^2 + \boldsymbol{s}(\mu)^T \boldsymbol{s}(\mu)''}{\|\boldsymbol{s}(\mu)\|} + \frac{(\boldsymbol{s}(\mu)^T \boldsymbol{s}(\mu)')^2}{\|\boldsymbol{s}(\mu)\|^2}$$

where

$$\boldsymbol{s}(\mu)'' = \mathbf{0}$$

- In fact, from

$$(\boldsymbol{H} + \mu \boldsymbol{I})\boldsymbol{s}(\mu)' = \boldsymbol{s}(\mu)$$

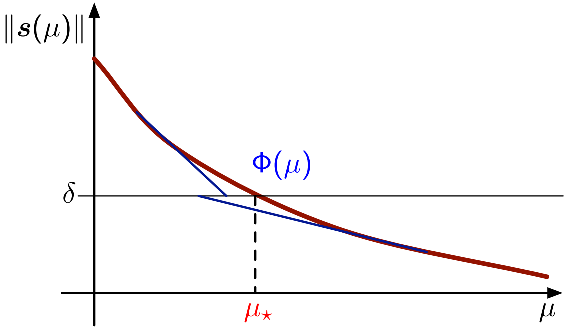
we have

$$\boldsymbol{H}\boldsymbol{s}(\mu)'' + \mu\boldsymbol{s}(\mu)'' + \boldsymbol{s}(\mu)' = \boldsymbol{s}(\mu)' \qquad \Rightarrow \qquad \boldsymbol{s}(\mu)'' = \mathbf{0}.$$

- Then for all $\mu \geq 0$ we have $\Phi''(\mu) > 0$.

The Newton approach (5/5)

- From $\Phi''(\mu) > 0$ we have that Newton step underestimates μ at each step.



- If we develop the vector $\boldsymbol{g}$ with the orthonormal bases given by the eigenvectors of $\boldsymbol{H}$ we have

$$\boldsymbol{g} = \sum_{i=1}^{n} \alpha_i \boldsymbol{u}_i$$

- Using this expression to evaluate $\boldsymbol{s}(\mu)$ we have

$$\boldsymbol{s}(\mu) = -(\boldsymbol{H} + \mu \boldsymbol{I})^{-1}\boldsymbol{g} = \sum_{i=1}^{n} \frac{\alpha_i}{\mu + \lambda_i}\boldsymbol{u}_i$$

$$\|\boldsymbol{s}(\mu)\| = \left(\sum_{i=1}^{n} \frac{\alpha_i^2}{(\mu + \lambda_i)^2}\right)^{1/2}$$

- This expression suggest to use as a model for $\Phi(\mu)$ the following expression

$$m_k(\mu) = \frac{\alpha_k}{\beta_k + \mu} - \delta$$

- The model consists of two parameter α_k and β_k. To set this parameter we can impose

$$m_k(\mu_k) = \frac{\alpha_k}{\beta_k + \mu_k} - \delta = \Phi(\mu_k)$$

$$m_k(\mu_k)' = -\frac{\alpha_k}{(\beta_k + \mu_k)^2} = \Phi(\mu_k)'$$

- solving for α_k and β_k we have

$$\alpha_k = -\frac{(\Phi(\mu_k) + \delta)^2}{\Phi(\mu_k)'} \qquad \beta_k = -\frac{\Phi(\mu_k) + \delta}{\Phi(\mu_k)'} - \mu_k$$

where

$$\Phi(\mu_k) = \|\boldsymbol{s}(\mu_k)\| - \delta \qquad \Phi(\mu_k)' = -\frac{\boldsymbol{s}(\mu_k)^T(\boldsymbol{H} + \mu_k\boldsymbol{I})^{-1}\boldsymbol{s}(\mu_k)}{\|\boldsymbol{s}(\mu_k)\|^2}$$

- Having α_k and β_k it is possible to solve $m_k(\mu) = 0$ obtaining

$$\mu_{k+1} = \frac{\alpha_k}{\delta} - \beta_k$$

- Substituting α_k and β_k the step become

$$\mu_{k+1} = \mu_k - \frac{\Phi(\mu_k)}{\Phi'(\mu_k)} - \frac{\Phi(\mu_k)^2}{\Phi'(\mu_k)\delta} = \mu_k - \frac{\Phi(\mu_k)}{\Phi'(\mu_k)}\left(1 + \frac{\Phi(\mu_k)}{\delta}\right)$$

- Comparing with the Newton step

$$\mu_{k+1} = \mu_k - \frac{\Phi(\mu_k)}{\Phi'(\mu_k)}$$

  we see that this method perform larger step by a factor $1 + \Phi(\mu_k)\delta^{-1}$.
- Notice that $1 + \Phi(\mu_k)\delta^{-1}$ converge to 1 as $\mu_k \to \mu_\star$. So that this iteration become the Newton iteration as μ_k becomes near the solution.

Algorithm (Exact trust region algorithm)

μ, $\boldsymbol{g}$, $\boldsymbol{H}$ *assigned;*

$\boldsymbol{s} \leftarrow (\boldsymbol{H} + \mu \boldsymbol{I})^{-1} \boldsymbol{g}$;

while $|\|\boldsymbol{s}\| - \delta| > \epsilon$ **do**

    — *compute the model*

    $\boldsymbol{s}' \leftarrow (\boldsymbol{H} + \mu \boldsymbol{I})^{-1} \boldsymbol{s}$;

    $\Phi \leftarrow \|\boldsymbol{s}\| - \delta$;

    $\Phi' \leftarrow -(\boldsymbol{s}^T \boldsymbol{s}')/(\boldsymbol{s}^T \boldsymbol{s})$

    $\alpha \leftarrow -(\Phi + \delta)^2 / \Phi'$;

    $\beta \leftarrow -(\Phi + \delta)/\Phi' - \mu$;

    — *update* μ *and* $\boldsymbol{s}$

    $\mu \leftarrow \dfrac{\alpha}{\delta} - \beta$;

    $\boldsymbol{s} \leftarrow (\boldsymbol{H} + \mu \boldsymbol{I})^{-1} \boldsymbol{g}$;

end while

Outline

The DogLeg approach (1/3)

- The computation of the μ such that $\|\boldsymbol{s}(\mu)\| = \delta$ of the exact trust region computation can be very expensive.
- An alternative was proposed by Powell:

  M.J.D. Powell
  A hybrid method for nonlinear equations
  in: Numerical Methods for Nonlinear Algebraic Equations
  ed. Ph. Rabinowitz, Gordon and Breach, pages 87-114, 1970.

  where instead of computing exactly the curve $\boldsymbol{s}(\mu)$ a piecewise linear approximation $\boldsymbol{s}_{dl}(\mu)$ is used in computation.
- This approximation also permits to solve $\|\boldsymbol{s}_{dl}(\mu)\| = \delta$ explicitly.

The DogLeg approach (2/3)

- Form the definition of $\boldsymbol{s}(\mu) = -(\boldsymbol{H} + \mu\boldsymbol{I})^{-1}\boldsymbol{g}$ it follows

$$\boldsymbol{s}(0) = -\boldsymbol{H}^{-1}\boldsymbol{g}, \qquad \lim_{\mu\to\infty} \frac{\boldsymbol{s}(\mu)'}{\|\boldsymbol{s}(\mu)'\|} = \frac{\boldsymbol{g}}{\|\boldsymbol{g}\|}$$

  i.e. the curve start from the Newton step and reduce to zero in the direction of the gradient step.

- The direction $-\boldsymbol{g}$ is a descent direction, so that a first piece of the piecewise approximation should be a straight line from $\boldsymbol{x}$ to the minimum of $m_k(\boldsymbol{x} - \lambda\boldsymbol{g})$. The minimum $\lambda_\star$ is found at

$$\lambda_\star = \frac{\|\boldsymbol{g}\|^2}{\boldsymbol{g}^T\boldsymbol{H}\boldsymbol{g}}$$

- Having reached the minimum if the $-\boldsymbol{g}$ direction we can now go to the point $\boldsymbol{x} + \boldsymbol{s}(0) = \boldsymbol{x} - \boldsymbol{H}\boldsymbol{g}$ with another straight line.

The DogLeg approach (3/3)

- We denote by

$$\boldsymbol{s}_g = -\boldsymbol{g}\frac{\|\boldsymbol{g}\|^2}{\boldsymbol{g}^T\boldsymbol{H}\boldsymbol{g}}, \qquad \boldsymbol{s}_n = -\boldsymbol{H}^{-1}\boldsymbol{g}$$

respectively the step due to the unconstrained minimization in the gradient direction and in the Newton direction.

- The piecewise linear curve connecting $\boldsymbol{x}+\boldsymbol{s}_n$, $\boldsymbol{x}+\boldsymbol{s}_g$ and $\boldsymbol{x}$ is the DogLeg curve[1] $\boldsymbol{x}_{dl}(\mu) = \boldsymbol{x} + \boldsymbol{s}_{dl}(\mu)$ where

$$\boldsymbol{s}_{dl}(\mu) = \begin{cases} \mu\boldsymbol{s}_g + (1-\mu)\boldsymbol{s}_n & \text{for } \mu \in [0,1] \\ (2-\mu)\boldsymbol{s}_g & \text{for } \mu \in [1,2] \end{cases}$$

[1] notice that $\boldsymbol{s}(\mu)$ is parametrized in the interval $[0,\infty]$ while $\boldsymbol{s}_{dl}(\mu)$ is parametrized in the interval $[0,2]$

Lemma

Consider the dogleg curve connecting $\boldsymbol{x} + \boldsymbol{s}_n$, $\boldsymbol{x} + \boldsymbol{s}_g$ and $\boldsymbol{x}$. The curve can be expressed as $\boldsymbol{x}_{dl}(\mu) = \boldsymbol{x} + \boldsymbol{s}_{dl}(\mu)$ where

$$\boldsymbol{s}_{dl}(\mu) = \begin{cases} \mu \boldsymbol{s}_g + (1-\mu)\boldsymbol{s}_n & \text{for } \mu \in [0,1] \\ (2-\mu)\boldsymbol{s}_g & \text{for } \mu \in [1,2] \end{cases}$$

for this curve if $\boldsymbol{s}_g$ is not parallel to $\boldsymbol{s}_n$ we have that the function

$$d(\mu) = \|\boldsymbol{x}_{dl}(\mu) - \boldsymbol{x}\| = \|\boldsymbol{s}_{dl}(\mu)\|$$

is strictly monotone decreasing, moreover the direction $\boldsymbol{s}(\mu)$ is a descent direction for all $\mu \in [0,2]$.

Proof. (1/5).

In order to have a unique solution to the problem $\|\boldsymbol{s}_{dl}(\mu)\| = \delta$ we must have that $\|\boldsymbol{s}_{dl}(\mu)\|$ is a monotone decreasing function:

$$\|\boldsymbol{s}_{dl}(\mu)\|^2 = \begin{cases} \mu^2 \boldsymbol{s}_g^2 + (1-\mu)^2 \boldsymbol{s}_n^2 + 2\mu(1-\mu)\boldsymbol{s}_g^T \boldsymbol{s}_n & \mu \in [0,1] \\ (2-\mu)^2 \boldsymbol{s}_g^2 & \mu \in [1,2] \end{cases}$$

To check monotonicity we take first derivative

$$\frac{\mathrm{d}}{\mathrm{d}\mu}\|\boldsymbol{s}_{dl}(\mu)\|^2$$

$$= \begin{cases} 2\mu \boldsymbol{s}_g^2 - 2(1-\mu)\boldsymbol{s}_n^2 + (2-4\mu)\boldsymbol{s}_g^T \boldsymbol{s}_n & \mu \in [0,1] \\ (2\mu-4)\boldsymbol{s}_g^2 & \mu \in [1,2] \end{cases}$$

$$= \begin{cases} 2\mu(\boldsymbol{s}_g^2 + \boldsymbol{s}_n^2 - 2\boldsymbol{s}_g^T \boldsymbol{s}_n) - 2\boldsymbol{s}_n^2 + 2\boldsymbol{s}_g^T \boldsymbol{s}_n & \mu \in [0,1] \\ (2\mu-4)\boldsymbol{s}_g^2 & \mu \in [1,2] \end{cases}$$

Proof. (2/5).

Notice that $(2\mu - 4) < 0$ for $\mu \in [1, 2]$ so that we need only to check that

$$2\mu(\boldsymbol{s}_g^2 + \boldsymbol{s}_n^2 - 2\boldsymbol{s}_g^T\boldsymbol{s}_n) - 2\boldsymbol{s}_n^2 + 2\boldsymbol{s}_g^T\boldsymbol{s}_n < 0 \qquad \text{for } \mu \in [0, 1]$$

Form the Cauchy-Schwartz inequality we have

$$\begin{aligned} \boldsymbol{s}_g^2 + \boldsymbol{s}_n^2 - 2\boldsymbol{s}_g^T\boldsymbol{s}_n &\geq \boldsymbol{s}_g^2 + \boldsymbol{s}_n^2 - 2\,\|\boldsymbol{s}_g\|\,\|\boldsymbol{s}_n\| \\ &= (\|\boldsymbol{s}_g\| - \|\boldsymbol{s}_n\|)^2 \geq 0 \end{aligned}$$

Then it is enough to check the inequality for $\mu = 1$

$$2(\boldsymbol{s}_g^2 + \boldsymbol{s}_n^2 - 2\boldsymbol{s}_g^T\boldsymbol{s}_n) - 2\boldsymbol{s}_n^2 + 2\boldsymbol{s}_g^T\boldsymbol{s}_n = 2\boldsymbol{s}_g^2 - 2\boldsymbol{s}_g^T\boldsymbol{s}_n$$

i.e. we must check $\boldsymbol{s}_g^2 - \boldsymbol{s}_g^T\boldsymbol{s}_n < 0$.

Proof. (3/5).

From the definition of $\boldsymbol{s}_g$ and $\boldsymbol{s}_n$ we have

$$\begin{aligned}\boldsymbol{s}_g^2 - \boldsymbol{s}_g^T \boldsymbol{s}_n &= \lambda_\star^2 \|\boldsymbol{g}\|^2 - \lambda_\star \boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g} \\ &= \lambda_\star \left[\frac{\|\boldsymbol{g}\|^2}{\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g}} \|\boldsymbol{g}\|^2 - \boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g} \right] \\ &= \frac{\lambda_\star}{\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g}} \left[\|\boldsymbol{g}\|^4 - (\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g})(\boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g}) \right]\end{aligned}$$

So that we must prove that

$$\|\boldsymbol{g}\|^4 < (\boldsymbol{g}^T \boldsymbol{H} \boldsymbol{g})(\boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g})$$

Proof. (4/5).

Expanding $\boldsymbol{g}$ by a set of orthonormal eigenvectors of $\boldsymbol{H}$ we have $\boldsymbol{g} = \sum_{i=1}^{n} \alpha_i \boldsymbol{u}_i$ and the the previous inequality becomes

$$\|\boldsymbol{g}\|^4 = \left(\sum_{i=1}^{n} \alpha_i^2\right)^2 = \left(\sum_{i=1}^{n} (\alpha_i \lambda_i^{1/2})(\alpha_i \lambda_i^{-1/2})\right)^2$$

$$\leq \left(\sum_{i=1}^{n} \alpha_i^2 \lambda_i\right)\left(\sum_{i=1}^{n} \alpha_i^2 \lambda_i^{-1}\right) = (\boldsymbol{g}\boldsymbol{H}\boldsymbol{g})(\boldsymbol{g}\boldsymbol{H}^{-1}\boldsymbol{g})$$

from the Cauchy–Schwartz inequality the previous inequality is strict unless

$$\alpha_i \lambda_i = c\alpha_i, \qquad i = 1, 2, \ldots, n$$

this means that $\lambda_i = c$ that for all $\alpha_i \neq 0$. This imply $\boldsymbol{H}^{-1}\boldsymbol{g} = c^{-1}\boldsymbol{g}$, i,e, Newton step and gradient step are parallel. But this is excluded in the lemma hypothesis.

Proof. (5/5).

To prove that $\boldsymbol{s}_{dl}(\mu)$ is a descent direction it is enough top notice that

- for $\mu \in [0, 1]$ the direction $\boldsymbol{s}_{dl}(\mu)$ is a convex combination of $\boldsymbol{s}_g$ and $\boldsymbol{s}_n$.
- for $\mu \in [1, 2)$ the direction $\boldsymbol{s}_{dl}(\mu)$ is parallel to $\boldsymbol{s}_g$.

so that it is enough to verify that $\boldsymbol{s}_g$ and $\boldsymbol{s}_n$ are descent direction.
For $\boldsymbol{s}_g$ we have

$$\boldsymbol{s}_g^T \boldsymbol{g} = -\lambda_\star \boldsymbol{g}^T \boldsymbol{g} < 0$$

For $\boldsymbol{s}_n$ we have

$$\boldsymbol{s}_n^T \boldsymbol{g} = -\boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g} < 0$$

□

Using the previous Lemma we can prove

Lemma

If $\|\boldsymbol{s}_{dl}(0)\| \geq \delta$ *then there is unique point* $\mu \in [0,2]$ *such that* $\|\boldsymbol{s}_{dl}(\mu)\| = \delta$.

Proof.

It is enough to notice that $\boldsymbol{s}_{dl}(2) = \mathbf{0}$ and that $\|\boldsymbol{s}_{dl}(\mu)\|$ is strictly monotonically descendent. □

The approximate solution of the constrained minimization can be obtained by this simple algorithm

1. if $\delta \leq \|\boldsymbol{s}_g\|$ we set $\boldsymbol{s}_{dl} = -\delta \boldsymbol{s}_g / \|\boldsymbol{s}_g\|$;
2. if $\delta \leq \|\boldsymbol{s}_n\|$ we set $\boldsymbol{s}_{dl} = \alpha \boldsymbol{s}_g + (1-\alpha)\boldsymbol{s}_n$; where α is the root in the interval $[0,1]$ of:

$$\alpha^2 \|\boldsymbol{s}_g\|^2 + (1-\alpha)^2 \|\boldsymbol{s}_n\|^2 + 2\alpha(1-\alpha)\boldsymbol{s}_g^T \boldsymbol{s}_n = \delta^2$$

3. if $\delta > \|\boldsymbol{s}_n\|$ we set $\boldsymbol{s}_{dl} = \boldsymbol{s}_n$;

Solving

$$\alpha^2 \|\boldsymbol{s}_g\|^2 + (1-\alpha)^2 \|\boldsymbol{s}_n\|^2 + 2\alpha(1-\alpha)\boldsymbol{s}_g^T \boldsymbol{s}_n = \delta^2$$

we have that if $\|\boldsymbol{s}_g\| \leq \delta \leq \|\boldsymbol{s}_n\|$ the root in $[0,1]$ is given by:

$$\Delta = \|\boldsymbol{s}_g\|^2 + \|\boldsymbol{s}_n\|^2 - 2\boldsymbol{s}_g^T \boldsymbol{s}_n = \|\boldsymbol{s}_g - \boldsymbol{s}_n\|^2$$

$$\alpha = \frac{\|\boldsymbol{s}_n\|^2 - \boldsymbol{s}_g^T \boldsymbol{s}_n - \sqrt{(\boldsymbol{s}_g^T \boldsymbol{s}_n)^2 - \|\boldsymbol{s}_g\|^2 \|\boldsymbol{s}_n\|^2 + \delta^2 \Delta}}{\Delta}$$

to avoid cancellation the computation formula is the following

$$\alpha = \frac{1}{\Delta} \frac{\|\boldsymbol{s}_n\|^4 - 2\boldsymbol{s}_g^T \boldsymbol{s}_n \|\boldsymbol{s}_n\|^2 + \|\boldsymbol{s}_g\|^2 \|\boldsymbol{s}_n\|^2 - \delta^2 \Delta}{\|\boldsymbol{s}_n\|^2 - \boldsymbol{s}_g^T \boldsymbol{s}_n + \sqrt{(\boldsymbol{s}_g^T \boldsymbol{s}_n)^2 - \|\boldsymbol{s}_g\|^2 \|\boldsymbol{s}_n\|^2 + \delta^2 \Delta}}$$

$$= \frac{\|\boldsymbol{s}_n\|^2 - \delta^2}{\|\boldsymbol{s}_n\|^2 - \boldsymbol{s}_g^T \boldsymbol{s}_n + \sqrt{(\boldsymbol{s}_g^T \boldsymbol{s}_n)^2 - \|\boldsymbol{s}_g\|^2 \|\boldsymbol{s}_n\|^2 + \delta^2 \|\boldsymbol{s}_g - \boldsymbol{s}_n\|^2}}$$

Algorithm (Computing DogLeg step)

dogleg$(\boldsymbol{s}_g,\ \boldsymbol{s}_n,\ \delta)$;

$$
\begin{aligned}
a &\leftarrow \|\boldsymbol{s}_g\|^2;\\
b &\leftarrow \|\boldsymbol{s}_n\|^2;\\
c &\leftarrow \|\boldsymbol{s}_g - \boldsymbol{s}_n\|^2;\\
d &\leftarrow (a+b-c)/2;\\
\alpha &\leftarrow \frac{b-\delta^2}{b-d+\sqrt{d^2-ab+\delta^2 c}};\\
\boldsymbol{s}_{dl} &\leftarrow \alpha\boldsymbol{s}_g + (1-\alpha)\boldsymbol{s}_n;
\end{aligned}
$$

return $\boldsymbol{s}_{dl}$;

References

J. Stoer and R. Bulirsch
Introduction to numerical analysis
Springer-Verlag, Texts in Applied Mathematics, **12**, 2002.

J. E. Dennis, Jr. and Robert B. Schnabel
Numerical Methods for Unconstrained Optimization and Nonlinear Equations
SIAM, Classics in Applied Mathematics, **16**, 1996.