Trust region Optimization

How principle of trust region opt. helps to grasp trust region policy opt.

Hyunin Lee

June 2024

1 Motivation

Trust region optimization is dedicated to solving non-convex optimization problem. Let's first start with any non-covex function $f : \mathbb{R}^n \to \mathbb{R}$. For any point $x_k \in \mathbb{R}^n$, our mission is computing the next iteration point x_{k+1} that guanratess a monotonic decrease, i.e. $f(x_{k+1}) < f(x_k)$. Let $x_{k+1} = x_k + p_k$. The taylor expansion around x_k is given as

$$f(x_k + p) = f_k + g_k^\top p + \frac{1}{2} p^\top \nabla^2 f(x_k + tp) p \tag{1}$$

where $f_k = f(x_k)$ and $g_k = \nabla f(x_k)$ and $t \in (0, 1)$ is a unknown constant. since $f(x_k + tp)$ is an unknown due to a constant t, let's approximate it as symmetric B_k, and let the approximate of $f(x_k + p)$ as m_k as follows.

$$m_k(p) = f_k + g_k^\top p + \frac{1}{2} p^\top B_k p \tag{2}$$

Then, we have the following approximated optimization problem with a bounded region.

$$\begin{aligned} \min_{p \in \mathbb{R}^n} \ & m_k(p) = f_k + g_k^\top p + \frac{1}{2} p^\top B_k p \\ \text{s.t.} \ & ||p|| \leq \Delta_k \end{aligned} \tag{3}$$

where $\Delta_k > 0$ is the trust-region radius. I would like to note that when 1) B_k is positive definite 2) $||B_k^{-1} g_k|| \leq \Delta_k$ then the Problem 3 have the unconstrained minimum as $p_k^B = -B_k^{-1} g_k$. We call p_k^B the full step. However, the problem is that this is too much computational expense, especially computing the $-B_k^{-1}$. This really necessaite to compute *approximate* solution.

2 Outline of Trust-region approach

One key graident of trust-region optimisation is the strategy for choosing trust region radius Δ_k. We base this choice on the agreement between the approimaximated model m_k and the objective function f. Given a step $p_k = \arg\min_{p \in \mathbb{R}^n} m_k(p)$, the minimum step of problem 3, we define the ratio as follows.

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)} \tag{4}$$

where we call the numerator as *actual reduction* and the denominator as the *predicted reduction.* I note that the denominoator is always non-negative since p_k is the argmin of m_k. Therefore, we have the following cases to determine how good our approximate m_k is as follows. Before, I note that if $\rho_k < 0$, then this means $f(x_k + p_k) > f(x_k)$, so the step must be rejected.

case1. When $\rho_k \approx 1$: this means m_k is a good approimator over this step. So it is safe to expand the trust region.

case2. When $\rho_k \approx 0$ and > 0: keep the trust region

case3. When $\rho_k \leq 0$: we should shrink the trust region.

Before moving on further, let's characterize the exact solutions of problem 3 by the following theorem.

Theorem 1. $p^\star \in \mathbb{R}^n$ is the global solution of the problem (3) if and only if $p^\star$ is feasible and there exists a scalar $\lambda \geq 0$ such that the following conditions are satisfied.

- $(B + \lambda I)p^\star = -g$
- $\lambda(\Delta - ||p^\star||) = 0$
- $(B + \lambda I)$ is a PSD matrix.

3 Algorithsm based on cauchy point

3.1 Cauchy point: sufficient reduction

Again, we are interested in finding an optimal solution to problem 3. First, it is enough to find an approximate solution p_k that lies within the trust region and gives a sufficient reduction. This *sufficient reduction* is quantified as *cauchy point*, denoted as point p_k^c.

Algorithm 1 (Cauchy point calculation)**.** Find vector $p_k^\star$ that solves linearized version of problem 3, that is,

$$p_k^s = \underset{p \in \mathbb{R}^n}{\arg\min} \, f_k + g_k^\top p \quad \text{s.t. } ||p|| \leq \Delta_k. \tag{5}$$

Then calculate the scalar $\tau_k > 0$ that minimizes $m_k(\tau_k p_k^s)$, that is

$$\tau_k = \underset{\tau \geq 0}{\arg\min} \, m_k(\tau p_k^s) \quad \text{s.t. } ||\tau p_k^s|| \leq \Delta_k \tag{6}$$

Then finally set $p_k^c = \tau_k p_k^s$.

The closed souiltion of cauchy point calculated by Algorihtm 1 is given as follows.

$$p_k^C = -\tau_k \frac{\Delta_k}{||g_k||} g_k \tag{7}$$

where

$$\tau_k = \begin{cases} 1 & \text{if } k_k^\top B_k g_k \leq 0 \\ \min(||g_k||^3/(\Delta_k g_k^\top B_k g_k), 1) & \text{otherwise} \end{cases}$$

3.2 Improving on the Cauchy points

What are some problems of using cauchy point p_k^c as the approximated solution of problem 3?

1. Cauchy point is merely steepest decent method. It is known that steepest descent performs poorly even if an optimal step length is used at each iteration.

2. Cauchy point does not utilized the B_k well. Rapid convergence can be expected when utilizing B_k as determining the direction of the step as well as its lenghth.

3.2.1 The dogleg method

First, note that this method assumptions B_k should be positive definite. Let's recall the problem 3 and rewrite the solution $p_k^\star$ as the function of radius Δ.

$$p_k^\star(\Delta) = \begin{cases} p^B, & \text{if } \Delta \geq ||p^B|| \\ -\Delta \frac{g}{||g||}, & \text{if } \Delta \text{ is small} \end{cases} \tag{8}$$

For the case of small Δ, the quadratic term of Problem 3 is negligible, so the solution is based on minimize the linearzed version of m_k. Since $p_k^\star$ is a function of radius Δ, it is easy to check that $p_k^\star$ is a curved trajectory. Then the dogleg method approximates is as the combination of two line segments P^U and P^B as follows.

$$\tilde{p}(\tau) = \begin{cases} \tau p^U, & 0 \leq \tau \leq 1 \\ p^U + (\tau - 1)(p^B - p^U), & 1 \leq \tau \leq 2 \end{cases} \tag{9}$$

where $p^U = -\frac{g^\top g}{g^\top B g} g$.

3.2.2 Two-diemnsional subspace minimization

It searches over the entire two-diemensional subspace spanned by p^U and p^B as follows.

$$\begin{aligned} \min_{p \in \mathbb{R}^n} \; & m_k(p) = f_k + g_k^\top p + \frac{1}{2} p^\top B_k p \\ \text{s.t. } & ||p|| \leq \Delta_k, \; p \in \text{span}[g_k, B_k^{-1} g_k] \end{aligned} \tag{10}$$

3.3 Global convergence

References