Chapter 7

Non-Monotone Trust-Region Methods for Set Optimization with Finitely many Vector-Valued functions as Set-Valued mapping

7.1 Introduction

The need for non-monotone trust region methods (NTRM) arise because of the known disadvantage of the conventional trust region method (TRM) which compels the objective function to monotonically decrease at every iteration. This requirement might be very restrictive and may hamper the chances and speed of convergence. Non-monotone schemes relaxes this enforcement over the objective functions by allowing for steps that may lead to increment in function values in some iterations. By doing this, non-monotone scheme can improve the probability and speed of convergence of monotone trust-region scheme. In the conventional literature on single and multi-objective optimization, there exist two type of non-monotone TRMs: max-type and average-type. Their main difference lie in the definition of reduction ratio and step-acceptance criteria. Max-type NTRM checks whether the current function value is smaller than maximum functions values over the previous M iterations, where M is a user defined parameter. On the other hand, average type compares the current function value to the average of the function values from all past iterations. Therefore, Max-type NTRM generates the sequence of maximum over function values, whereas Average NTRM generates the sequence of average over function values. In comparison, TRM generates a sequence of current function values.

Numerous studies have utilized non-monotone adjustments to trust region methods for both single-objective and multi-objective optimization problems. Deng et al. were the first to implement such a modification for single-objective optimization. Since then, many other studies have followed, further exploring the performance of NMTR for single objective optimization, for instance, sun [171], Chen et al. [29], Ahookhosh et al. [2], Macial et al. [135], Mo et al. [140] etc. Non-monotone trust region schemes have also been successfully implemented for multi-objective optimization problems. For example, [154] enhanced the trust region method by developing a non-monotone version, demonstrating significant improvements in number of iterations, computation time, and the number of subproblems that needed to be solved for multi-objective problems. Following this, Ding et al. [42] built upon [154] by developing an adaptive non-monotone trust-region method to tackle a specific type of multi-objective non-linear bi-level optimization (MNBLO) problem. They achieved this by taking the convex combination of the current iteration's function value and the maximum over the function values from previous iterations. Additionally, Ghalavand et al. [61] expanded the max-based adaptive method from [42], which was limited to MNBLO problems, to any general multi-objective scenario. They also investigated the average-type adaptation of the adaptive trust region method.

7.2 Motivation

In monotone TRM, the step acceptance criteria is restricted to accept only those steps that reduces the function values compared to the immediate previous iteration. This limitation is quite stringent, as evidenced by observations in both single-objective [2, 140] and multi-objective optimization [61, 154]. Such a restriction may limit the potential for convergence and could also slow down the convergence rate. For the case of single and multi objective optimization, researchers have developed non-monotone modifications of TRM which have been shown to be significantly better than TRM for various types of objective functions. Following this trend, we should expect similar performance gains for the case of set optimization as well. However, in the literature, because of the challenges and difficulties in set optimization, NMTR for set optimization have not been developed yet. This motivates us to implement and study the performance of Max-NTRM and Avg-NTRM for set optimization.

7.3 Contributions

In this chapter, we present two new variants of non-monotone trust region methods: the max-type non-monotone trust region method (Max-NTRM) and the average-type non-monotone trust region method (Avg-NTRM), designed for discrete non-convex unconstrained set optimization problems (SOP). These two schemes lead to their own step acceptance criterion (based on their defined reduction ratios) where Max-NTRM considers the maximum over functions values from the last M iterations, and Avg-NTRM takes an exponentially weighted moving average of function values till the current iteration. The main contributions of this chapter can be summarized as follows:

(i) We introduce non-monotone Max-NTRM and Avg-NTRM algorithms aimed at identifying critical points for a specific type of set optimization problem. We demonstrate the well-definedness of each algorithm step.

(ii) We establish the global convergence of the two proposed methods to provide theoretical guarantees.

(iii) We conduct numerical experiments on 20 example set optimization problems to compare the performance of Max-NTRM and Avg-NTRM against the monotone TRM method given Chapter 6.

(iv) We generate a performance profile for the three methods using the profiling methodology outlined in [43], focusing on four performance metrics: convergence rate, number of iterations, computation time, and average step size.

(v) Based on the analysis of the performance profile, we observe that Max-NTRM generally excels across all metrics. However, Avg-NTRM also demonstrates very strong performance and, in cases where it does not outperform TRM, remains very competitive.

7.4 Non-monotone trust-region method for set optimization

In this section, we propose Max-NTRM and Avg-NTRM non-monotone trust region methods for set optimization using the (monotone) trust region method, given in 6, as the basis. Since our schemes are refinement of TRM, most of the standard steps such as calculation of partition P_{x_k}, the choice of a^k from that P_{x_k} etc. are similar. The main modification happens in the step of reduction ratio calculation using which we show that our proposed method performs better than TRM. As a quick recap, we, in Chapter 6 proved that their trust region algorithm for SOP generates a sequence of iterates $\{x_k\}$ that finally converges to the critical point of the SOP. Their necessary condition of criticality states that if a point x_k is a critical point of ($\mathcal{SOP}^l_K$), then $\theta(x_k) = 0$. The function $\theta : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$\theta(x) = \min_{(a,s)\in P_x \times \mathcal{B}} \Theta_x(a, s), \tag{7.1}$$

where, $\Theta_x : P_x \times \mathcal{B} \to \mathbb{R}$, for any given $x \in \mathbb{R}^n$, is given by

$$\Theta_x(a,s) = \max_{j\in[\omega(x)]} \left\{\Delta_{-K}\left(\nabla f^{a_j}(x)^\top s + \tfrac{1}{2}s^\top \nabla^2 f^{a_j}(x)s\right), \Delta_{-K}(\nabla f^{a_j}(x)^\top s)\right\}. \tag{7.2}$$

Here, P_x is the partition set at x and $\mathcal{B} = \{s \in \mathbb{R}^n : \|s\| \leq \Omega_{\max}\}$ with $\Omega_{\max}$ as the maximum allowed trust-region step-size. At every iteration k, an $a^k = \left(a^k_1, a^k_2, \ldots, a^k_{\omega_k}\right)$ is selected from the partition set P_k by solving

$$(a^k, s_k) \in \operatorname*{argmin}_{(a,s)\in P_k\times\mathcal{B}_k} \max_{j\in[\omega_k]} \left\{\Delta_{-K}\left(\nabla f^{a_j}(x_k)^\top s + \tfrac{1}{2}s^\top \nabla^2 f^{a_j}(x_k)s\right), \Delta_{-K}\left(\nabla f^{a_j}(x_k)^\top s\right)\right\}, \tag{7.3}$$

such that $\theta(x_k) = \Theta_{x_k}(a^k, s_k)$. Then, using a^k, the corresponding vector optimization problem is defined as

$$(\preceq_{\widetilde{K}}) \quad \min_{x\in\mathbb{R}^n} \widetilde{f}^{a^k}(x),\ a^k \in P_k,. \tag{$\mathcal{VOP}_{a^k}(x_k)$}$$

The possible step s_k inside the trust-region $\mathcal{B}_k$ is computed using the model function $\widetilde{m}^{a^k}$ and its corresponding subproblem given by

$$\left.\begin{aligned} &\min && t \\ &\text{subject to} && \Delta_{-K}\left(\nabla f^{a^k_j}(x_k)^\top s + \tfrac{1}{2}s^\top \nabla^2 f^{a^k_j}(x_k)s\right) - t \leq 0,\ j = 1,2,\ldots,\omega_k, \\ &&& \Delta_{-K}\left(\nabla f^{a^k_j}(x_k)^\top s\right) - t \leq 0,\ j = 1,2,\ldots,\omega_k, \\ &&& \|s\| \leq \Omega_k. \end{aligned}\right\} \tag{7.4}$$

Using Theorem 7.5.2, the authors show that step s_k of ($\mathcal{VOP}_{a^k}(x_k)$), if accepted, is also the accepted step for ($\mathcal{SOP}^l_K$) at x_k. For monotone TRM, the reduction ratios $\rho_k^{a^k_j}$, for all $a^k_j \in a_k$ and $j \in [\omega_k]$, used in the step acceptance criterion are given by

$$\rho_k^{a^k_j} = \frac{\text{actual function reduction}}{\text{predicted reduction}} = -\frac{\Delta_{-K}(f^{a^k_j}(x_k+s_k) - f^{a^k_j}(x_k))}{\Delta_{-K}(m^{a^k_j}(0) - m^{a^k_j}(s_k))}. \tag{7.5}$$

Here, only the function values at the current iteration $f^{a^k_j}(x_k)$ is utilised to make a decision about the computed step s_k.

In this paper, we modify these reduction ratios so that it takes into account not just the function values from current iteration but also from previous iterations. Based on the way in which such information from previous iterations is combined gives rise to Max-NTRM and Avg-NTRM schemes. For Max-NTRM, drawing motivation from Sun's [171] and Ramirez et al's [154] methods for single-objective and multi-objective optimization, respectively, we define the reduction ratios $\rho_k^{a^k_j}$, for all $a^k_j \in a_k$ and $j \in [\omega_k]$, as

$$\rho_k^{a_j^k} = -\frac{\Delta_{-K}\left(f^{a_j^k}(x_k+s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}\right)}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} \tag{7.6}$$

where, for all $i \in [p]$,

$$f^{i,r}(x_{l^{i,r}(k)}) = \begin{cases} \max_{0\le q\le N_k} f^{i,r}(x_{k-q}) & \text{if } a^k = a^{k-1}, \cdots, = a^{k-q}, \cdots, = a^{k-N_k} \\ f^{i,r}(x_k) & \text{otherwise.} \end{cases} \tag{7.7}$$

Here, $N_k = \min\{N_{k-1}+1, \hat{N}\}$ is the number of previous iterations over which the maximum of the function values is calculated. $N_0 = 0$, and $\hat{N}$ is the largest value N_k can take i.e. the maximum number of past iterations we can look back. The above means that, when the set a^k of the current and last N_k iterations are same, we take the maximum of their function values over the last N_k iterations to calculate the set of reduction ratios $\rho_k^{a_j^k}$. If any a^k from previous N_K iterations is different, then, the algorithm reduces to the case of monotone TRM where we simply take the current function values to calculate $\rho_k^{a_j^k}$. Note that, for Max-NTRM, $\rho_k^{a_j^k} > 0$, means $\Delta_{-K}(f^{a_j^k}(x_k+s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}) < 0$ and $f^{a_j^k}(x_k+s_k) \prec_K (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}$, i.e., for acceptance, s_k does not need to be along the descent direction of $f^{a_j^k}$ at x_k like TRM. This allows a non-monotonic freedom to the function values so that they can increase for some iterations if necessary. Thus, $-\Delta_{-K}(f^{a_j^k}(x_k+s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]})$ is the non-monotone reduction of the actual vector-valued function $f^{a_j^k}$ due to the step s_k, and the predicted reduction $\Delta_{-K}(m^{a_j^k}(0) - m^{a_j^k}(s_k))$ is the same as in Chapter 6. Before we move forward, for clarity of notations, we define the following

$$\begin{aligned} F(x_{l(k)}) = \left\{f^i(x_{l^i(k)})\right\}_{i\in[p]} &= \left\{\left(f^{i,r}(x_{l^{i,r}(k)})\right)_{r\in[m]}\right\}_{i\in[p]} \\ &= \left\{\left(f^{i,1}(x_{l^{i,1}(k)}), f^{i,2}(x_{l^{i,2}(k)}), \ldots, f^{i,m}(x_{l^{i,m}(k)})\right)\right\}_{i\in[p]}. \end{aligned} \tag{7.8}$$

This maximum-based non-monotone scheme, however, can be disadvantageous in some cases. For example, when objective values at the current iteration are very good, it would be more sensible to utilise them, however, Max-NTRM completely ignores them to instead consider the maximum objective values over past iterations [2]. To address this particular issue, in the literature, various average-based non-monotone schemes have been proposed which, instead of the maximum, considers a weighted average of the function values from previous iterations. Therefore, motivated by Mo et al.'s [140] and Ghalavand et al's [61] average-based NTRM schemes for single-objective and multi-objective cases, respectively, we define the Avg-NTRM for set optimization who's reduction ratios $\rho^{a_j^k}$ are defined as

$$\rho_k^{a_j^k} = \frac{-\Delta_{-K}(f^{a_j^k}(x_k+s_k) - (C_k^{a_j^k,r})_{r\in[m]})}{\Delta_{-K}(m^{a_j^k}(0) - m^{a_j^k}(s_k))}, \tag{7.9}$$

where, for all $i \in [p]$,

$$C_k^{i,r} = \begin{cases} \frac{\mu_{k-1}q_{k-1}}{q_k} C_{k-1}^{i,r} + \frac{1}{q_k} f^{i,r}(x_k) & \text{if } a^k = a^{k-1}, \cdots, = a^0 \\ f^{i,r}(x_k) & \text{otherwise,} \end{cases} \tag{7.10}$$

where,

$$q_k = \begin{cases} 1 & k = 0 \\ \mu_{k-1}q_{k-1} + 1 & k \geq 1, \end{cases}$$

for $\mu_k \in [\mu_{\min}, \mu_{\max}]$, with $\mu_{\min} \in [0,1)$ and $\mu_{\max} \in [\mu_{\min}, 1)$. In other words, when the set a^k of the current and all previous iterations are same, we take the average of their function values to calculate ratios $\rho_k^{a_j^k}$. If any of the a^k from all previous iterations is different, then, the algorithm reduces to monotone TRM which simply considers the current function values. For Avg-NTRM, when step s_k is accepted, we get the condition that $f^{a_j^k}(x_k+s_k) \prec_K (C_k^{a_j^k,r})_{r\in[m]}$.

Therefore, Max-NTRM and Avg-NTRM, only needs the sequence $\{f^i(x_{l^i(k)})\}$ and $\{C_k^i\}$, respectively, to be monotonic, thus allowing the sequence $\{f^i(x_k)\}$ to be non-monotonic. For the special case of $\hat{N} = 0$, $f^i(x_{l^i(k)}) = f^i(x_k)$ and the reduction ratio of Max-NTRM reduces to (7.5). Similarly, for $\mu_k = 0$ for each k, $C_k^i = f^i(x_k)$ and the reduction ratio of Avg-NTRM reduces to (7.5). Based on the two reduction ratios (7.6) or (7.9), we propose the Max-NTRM and Avg-NTRM algorithms which are given in Algorithm 4 and Algorithm 5, respectively. The trust region radius update rule and stopping criterion for these two algorithms is same as that of monotone TRM algorithm of Chapter 6. The algorithm is stopped when it finds a point of criticality. As per Chapter 6, a critical point our considered ($\mathcal{SOP}_K^l$) is characterized by the optimal values $\theta(x_k)$ (in the image space) and $s(x_k)$ (in the variable space) of the subproblem (7.4), which are defined as

$$\theta(x_k) = \min_{s\in\mathcal{B}_k} \max_{j\in[\omega_k]} \left\{\Delta_{-K}\left(\nabla f^{a_j^k}(x_k)^\top s + \tfrac{1}{2} s^\top \nabla^2 f^{a_j^k}(x_k) s\right), \Delta_{-K}\left(\nabla f^{a_j^k}(x_k)^\top s\right)\right\} \tag{7.11}$$

and

$$s(x_k) = \underset{s\in\mathcal{B}_k}{\operatorname{argmin}} \max_{j\in[\omega_k]} \left\{ \Delta_{-K}\left(\nabla f^{a_j^k}(x_k)^\top s + \tfrac{1}{2}s^\top \nabla^2 f^{a_j^k}(x_k)s\right), \Delta_{-K}\left(\nabla f^{a_j^k}(x_k)^\top s\right)\right\}. \tag{7.12}$$

Next, similar to the Proposition 3.2 of Chapter 6, we show the interdependence between reduction ratios and accepted step criterion in the following proposition.

Proposition 7.4.1 *Let x_k be a noncritical point of ($\mathcal{SOP}_K^l$). Then, for any $j \in [\omega_k]$, a non-monotone trust region step s_k satisfies $f_k^{a_j^k}(x_k + s_k) \prec_K (f^{a_j^k,r}(x_{l_j^r(k)}))_{r\in[m]}$ (Max-NTRM) or $f_k^{a_j^k}(x_k + s_k) \prec_K (C_k^{a_j^k,r}(x_k))_{r\in[m]}$ (Avg-NTRM) if and only if $\rho_k^{a_j^k} > 0$.*

Proof: Let s_k satisfy $f^{a_j^k}(x_k + s_k) \prec_K (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}$ (Max-NTRM) or $f^{a_j^k}(x_k + s_k) \prec_K (C_k^{a_j^k,r})_{r\in[m]}$ (Avg-NTRM). Then, $\Delta_{-K}(f_k^{a_j^k}(x_k + s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}) < 0$ (Max-NTRM) or $\Delta_{-K}(f_k^{a_j^k}(x_k+s_k)-(C_k^{a_j^k,r})_{r\in[m]}) < 0$ (Avg-NTRM). As s_k is identified by solving (7.4), we get

$$\begin{aligned}\max_{j\in[\omega_k]} \Delta_{-K}(m_k^{a_j^k}(s_k)) &\le \max_{j\in[\omega_k]} \left\{\Delta_{-K}(m_k^{a_j^k}(s_k)),\ \Delta_{-K}(\nabla f_k^{a_j^k}(x_k)^\top s_k)\right\}\\ &\le \max_{j\in[\omega_k]} \left\{\Delta_{-K}(m_k^{a_j^k}(0)),\ \Delta_{-K}(\nabla f_k^{a_j^k}(x_k)^\top 0)\right\} = 0,\end{aligned}$$

i.e., $\Delta_{-K}(m_k^{a_j^k}(s_k)) \le 0$. Thus,

$$\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k)) \ge \Delta_{-K}(m_k^{a_j^k}(0)) - \Delta_{-K}(m_k^{a_j^k}(s_k)) \ge 0. \tag{7.13}$$

For a noncritical point x_k, from Corollary 6.6.1, we get $\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k)) > 0$. Thus,

$$\begin{aligned}\rho_k^{a_j^k} &= -\frac{\Delta_{-K}(f^{a_j^k}(x_k + s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]})}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} > 0 \text{ (Max-NTRM)} \qquad (7.14)\\ &= -\frac{\Delta_{-K}(f^{a_j^k}(x_k + s_k) - ((C_k^{a_j^k,r})_{r\in[m]})}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} > 0 \text{ (Avg-NTRM)}\ . \qquad (7.15)\end{aligned}$$

Conversely, suppose $\rho_k^{a_j^k} > 0$. Then, $\Delta_{-K}(f^{a_j^k}(x_k + s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}) < 0$ (Max-NTRM) or $\Delta_{-K}(f^{a_j^k}(x_k + s_k) - (C_k^{a_j^k,r})_{r\in[m]}) < 0$ (Avg-NTRM), and hence $f^{a_j^k}(x_k + s_k) \prec_K (f^{a_j^k,r}(x_{l^{j,r}(k)})_{r\in[m]}$ (Max-NTRM) or $f^{a_j^k}(x_k+s_k) \prec_K (C_k^{a_j^k,r})_{r\in[m]}$ (Avg-NTRM). $\square$

Algorithm 4 Max-NTRM algorithm for solving ($\mathcal{SOP}^{l}_{K}$)

1: *Initialization and Inputs*
Inputs to problem ($\mathcal{SOP}^{l}_{K}$): $f^i : \mathbb{R}^n \to \mathbb{R}^m$, $i = 1, 2, \ldots, p$
Initialize parameters: iteration counter $k = 0$, initial point x_0, initial trust region radius Ω_0, maximum trust region radius $\Omega_{\max}$, threshold parameters $\eta_1, \eta_2 \in (0, 1)$, fractions $\gamma_1, \gamma_2 \in (0, 1)$ to shrink the trust region radius, tolerance value ϵ for stopping criterion, number of past iterations to consider for max calculation $N_0 = 0$, and upper bound $\hat{N}$.

2: *Find minimal elements*
Calculate $M_k = \operatorname{Min}(F(x_k), K) = \{r_1, r_2, \ldots, r_{w_k}\}$.
Calculate the partition set $P_k = P_{x_k} = I_{r_1} \times I_{r_2} \times \cdots \times I_{r_{\omega_k}}$.
Compute $p_k = |P_k|$ and $\omega_k = |\operatorname{Min}(F(x_k), K)|$.

3: *Selection of an 'a^k' from P_k*
Choose an element $a^k = (a^k_1, a^k_2, \ldots, a^k_{\omega_k}) \in P_k$ as per

$$(a^k, s_k) \in \underset{(a,s)\in P_k \times \mathcal{B}_k}{\operatorname{argmin}} \max_{j\in[\omega_k]} \left\{ \Delta_{-K}\left(\nabla f^{a_j}(x_k)^\top s + \tfrac{1}{2} s^\top \nabla^2 f^{a_j}(x_k) s \right), \Delta_{-K}\left(\nabla f^{a_j}(x_k)^\top s \right) \right\}.$$

4: *Definition of Model function*
For all $j \in [\omega_k]$, calculate the model functions $m_k^{a_j^k}$ as

$$m_k^{a_j^k}(s) = \nabla f^{a_j^k}(x_k)^\top s + \tfrac{1}{2} s^\top \nabla^2 f^{a_j^k}(x_k) s, \quad \|s\| \le \Omega_k.$$

5: *Step computation*
Calculate (t_k, s_k) by solving subproblem (3.11) as in Algorithm 3 of Chapter 6.

6: *Stopping criterion*
If $|t_k| < \epsilon$, terminate the algorithm and output x_k as a critical point of ($\mathcal{SOP}^{l}_{K}$).
Else, go to next step.

7: *Reduction ratio* Compute reduction ratio as per (7.6):

$$\rho_k^{a_j^k} = -\frac{\Delta_{-K}\left(f^{a_j^k}(x_k + s_k) - (f^{a_j^k, r}(x_{l^{j,r}(k)}))_{r\in[m]} \right)}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} \text{ for all } j \in [\omega_k].$$

8: *Step acceptance criterion*
If $\rho_k^{a_j^k} \ge \eta_1$ for all $j \in [\omega_k]$, then successful step. Set $x_{k+1} = x_k + s_k$.
Else If $\rho_k^{a_j^k} < \eta_1$ for at least one j, then unsuccessful step. Set $x_{k+1} = x_k$.

9: *Update trust region radius*
Select

$$\Omega_{k+1} \in \begin{cases} (\gamma_2 \Omega_k, \Omega_k], & \text{if } \eta_1 \le \rho_k^{a_j^k} \ \forall\, j \in [\omega_k] \text{ and } \exists\, l \in [\omega_k] \text{ such that } \rho_k^{a_l^k} < \eta_2 \ (\textbf{Successful}) \\ (\Omega_k, \infty), & \text{if } \eta_2 \le \rho_k^{a_j^k} \ \forall\, j \in [\omega_k], \ (\textbf{Very successful}) \\ [\gamma_1 \Omega_k, \gamma_2 \Omega_k], & \text{if } \exists\, l \in [\omega_k] \text{ such that } \rho_k^{a_l^k} \le \eta_1 \ (\textbf{Unsuccessful}). \end{cases}$$

10: *Update N_k and go to next iteration*
Set $N_{k+1} = \min\{N_k + 1, \ \hat{N}\}$.
Set $k = k + 1$ and go to Step 2:.

Algorithm 5 Average-NTRM algorithm for solving ($\mathcal{SOP}^l_K$)

1: *Initialization and Inputs*
Inputs to problem ($\mathcal{SOP}^l_K$): $f^i : \mathbb{R}^n \to \mathbb{R}^m$, $i = 1, 2, \ldots, p$
Initialize parameters: iteration counter $k = 0$, initial point x_0, initial trust region radius Ω_0, maximum trust region radius $\Omega_{\max}$, threshold parameters $\eta_1, \eta_2 \in (0, 1)$, fractions $\gamma_1, \gamma_2 \in (0, 1)$ to shrink the trust region radius, tolerance value ϵ for stopping criterion, $(C_0^{i,r})_{r\in[m]} = (f^{i,r}(x_0))_{r\in[m]}$ $\forall\ i \in [p]$, $q_0 = 1$, and take $0 \leq \mu_{\min} \leq \mu_{\max} < 1$, $\mu_0 \in [\mu_{\min}, \mu_{\max}]$.

2: *Find minimal elements*
Calculate $M_k = \text{Min}(F(x_k), K) = \{r_1, r_2, \ldots, r_{w_k}\}$.
Calculate the partition set $P_k = P_{x_k} = I_{r_1} \times I_{r_2} \times \cdots \times I_{r_{\omega_k}}$.
Compute $p_k = |P_k|$ and $\omega_k = |\text{Min}(F(x_k), K)|$.

3: *Selection of an 'a^k' from P_k*
Choose an element $a^k = (a^k_1, a^k_2, \ldots, a^k_{\omega_k}) \in P_k$ as per

$$(a^k, s_k) \in \underset{(a,s)\in P_k\times\mathcal{B}_k}{\text{argmin}} \max_{j\in[\omega_k]} \left\{\Delta_{-K}\left(\nabla f^{a_j}(x_k)^\top s + \tfrac{1}{2}s^\top\nabla^2 f^{a_j}(x_k)s\right), \Delta_{-K}\left(\nabla f^{a_j}(x_k)^\top s\right)\right\}.$$

4: *Definition of Model function*
For all $j \in [\omega_k]$, calculate the model functions $m_k^{a^k_j}$ as

$$m_k^{a^k_j}(s) = \nabla f^{a^k_j}(x_k)^\top s + \tfrac{1}{2}s^\top\nabla^2 f^{a^k_j}(x_k)s, \quad \|s\| \leq \Omega_k.$$

5: *Step computation*
Calculate (t_k, s_k) by solving subproblem (3.11) as in Algorithm 3 of Chapter 6.

6: *Stopping criterion*
If $|t_k| < \epsilon$, terminate the algorithm and output x_k as a critical point of ($\mathcal{SOP}^l_K$).
Else, go to next step.

7: *Reduction ratio*
Compute reduction ratio as per (7.9):

$$\rho_k^{a^k_j} = -\frac{\Delta_{-K}\left(f^{a^k_j}(x_k + s_k) - (C_k^{a^k_j,r})_{r\in[m]}\right)}{\Delta_{-K}(m^{a^k_j}(0) - m^{a^k_j}(s_k))},$$

8: *Step acceptance criterion*
If $\rho_k^{a^k_j} \geq \eta_1$ for all $j \in [\omega_k]$, then successful step. Set $x_{k+1} = x_k + s_k$.
Else If $\rho_k^{a^k_j} < \eta_1$ for at least one j, then unsuccessful step. Set $x_{k+1} = x_k$.

9: *Update trust region radius*
Select

$$\Omega_{k+1} \in \begin{cases} (\gamma_2\Omega_k, \Omega_k], & \text{if } \eta_1 \leq \rho_k^{a^k_j}\ \forall\ j \in [\omega_k] \text{ and } \exists\ l \in [\omega_k] \text{ such that } \rho_k^{a^k_l} < \eta_2\ (\textbf{Successful}) \\ (\Omega_k, \infty), & \text{if } \eta_2 \leq \rho_k^{a^k_j}\ \forall\ j \in [\omega_k],\ (\textbf{Very successful}) \\ [\gamma_1\Omega_k, \gamma_2\Omega_k], & \text{if } \exists\ l \in [\omega_k] \text{ such that } \rho_k^{a^k_l} \leq \eta_1\ (\textbf{Unsuccessful}). \end{cases}$$

10: *Go to next iteration*
Set $k = k + 1$ and go to Step 2:.

7.4.1 Well-definedness of Algorithm 4 and Algorithm 5

The well-definedness of our proposed algorithms depends on their respective Step 3:, Step 5:, Step 6:, Step 7:, and Step 8:. Among them, Step 3:, Step 5:, and Step 6: are same as those in TRM algorithm, whose well-definedness is already given in subsection 3.6 of Chapter 6. For Step 7: and Step 8:, the well-definedness is presented next.

Step 7: calculates the respective reduction ratios of the two algorithms. With s_k as the solution of subproblem (7.4), we get the predicted reduction term $\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))$ in the denominator of reduction ratio to be positive. Also, for s_k to be accepted, we have $f^{a_j^k}(x_k + s_k) \prec_K (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}$ (MAx-NTRM) or $f^{a_j^k}(x_k + s_k) \prec_K (C^{a_j^k,r})_{r\in[m]}$ (Avg-NTRM). This, thus, creates an interdependence between reduction ratios $\rho_k^{a_j^k}$ and the step acceptance criteria as shown in Proposition 7.4.1. This interdependence causes Step 7: of Max-NTRM and Avg-NTRM to be analogous to the step acceptance criterion of the conventional Max-NTRM and Avg-NTRM, respectively. Thus, the choice of $\rho_k^{a_j^k}$ for both Algorithms is well-defined.

Step 8: checks whether the computed step s_k can be accepted or not. From Corollary 7.5.2 (presented later) and Proposition 7.4.1 tells us that if there exists a step s_k (obtained from solving ($\mathcal{VOP}_{a^k}(x_k)$)) that satisfies $\rho_k^{a_j^k} > 0$ for all $j \in [\omega_k]$, then this step can be accepted for ($\mathcal{SOP}^l_K$) and we get $f^i(x_k + s_k) \prec_K (f^{i,r}(x_{l^{j,r}(k)}))_{r\in[m]}$ (Max-NTRM) or $f^i(x_k + s_k) \prec_K (C^{i,r})_{r\in[m]}$ (Avg-NTRM) for all $i \in [p]$. Thus, Step 8: is well-defined.

Finally, Theorem 7.1 shows that Algorithm 4 and Algorithm 5 generates the sequence $\{x_k\}$ in such a way that $F(x_{l(k+1)}) \preceq^l_K F(x_{l(k)})$ and $\{(C^{i,r}_{k+1})_{r\in[m]}\}_{i\in[p]} \preceq^l_K \{(C^{i,r}_k)_{r\in[m]}\}_{i\in[p]}$, respectively. This is in line with the core idea of conventional max-type and avg-type non-monotone trust-region methods, i.e., the sequence of maximum of function values from past iterations or the sequence of average of function values from past iterations decrease monotonically. Therefore, Algorithm 4 and 5 are well-defined.

7.5 Global convergence analysis

In this section, we prove that both Max-NTRM and Avg-NTRM converge to a critical point for ($\mathcal{SOP}^l_K$). For this, we first make a few assumptions on the considered set-valued function $F = \{f^i\}_{i\in[p]}$. These assumptions are very common in single and multi-objective non-monotone trust-region schemes.

Assumption 7.5.1 *Function f^i is continuously differentiable for all $i \in [p]$.*

Assumption 7.5.2 *Hessian of function f^i, for all $i \in [p]$, is uniformly bounded. This means that there exists a $\mathcal{K}_1$ such that*

$$\|\nabla^2 f^i(x)\| \leq \mathcal{K}_1.$$

Assumption 7.5.3 *There exists a constant vector M_1 whose every coordinate is positive real number, such that*

$$s^\top \nabla^2 f^i(x) s \preceq_K M_1 \|s\|^2 \text{ for all } i \in [p].$$

Assumption 7.5.4 *Level set of function F, $\mathcal{L}_c = \{x \in \mathbb{R}^n : F(x) \preceq_K^l F(x_0)\}$, is bounded.*

Assumption 7.5.5 *Function f^i, for all $i \in [p]$, is bounded below.*

Assumption 7.5.6 *There exist $\mathcal{K}_3 > 0$ such that, for all $i \in [p]$,*

$$\|\nabla f^i(x)\| \leq \mathcal{K}_3.$$

Next, we present a series of lemmas, corollaries, and theorems which ultimately lead to the convergence proof of our algorithms. We present Lemma 7.1, to be used in Corollary 7.5.1, to show a bound on the sufficient decrease of the model function m^{a^k}. Next, we present Lemma 7.2, followed by Corollary 7.5.1 of Chapter 6 which will be useful later. Then, in Theorem 7.1, we show that the sequence of function values generated by the two algorithms are monotonically decreasing. Next, we present a Corollary 7.5.2 that connects the accepted step of ($\mathcal{VOP}_{a^k}(x_k)$) to accepted step of ($\mathcal{SOP}^l_K$). Next, Theorem 7.2 ensures that the algorithms always generate a successful step after finite number of unsuccessful steps. Next, in Lemma 7.5.1, we show that the sequence of functions generated by the two algorithms admits a limit. Finally, in Theorem 7.3, we show that the sequence of iterates generated by the two algorithms converge to critical point of ($\mathcal{SOP}^l_K$).

Now, we present a lemma which will be later used in Corollary 7.5.1 to show the sufficient decrease of the model m^{a^k} for our considered ($\mathcal{VOP}_{a^k}(x_k)$) in terms of the optimal value $\theta(x_k)$ of the subproblem (7.4). This corollary will then be used for proving the global convergence of our algorithms.

Lemma 7.1 *For an iteration $k \in \mathbb{N}$, let x_k be a non-critical point for ($\mathcal{VOP}_{a^k}(x_k)$). Further, let $f^i \in C^2(\mathbb{R}^n, \mathbb{R}^m)$, and let $\nabla f^i(x)$ be positive definite for all $i \in [p]$. Additionally, set $T = -\Delta_{-K}(-M) > 0$, where M fulfills (7.5.3). Then, s_k, obtained by solving subproblem (7.4), satisfies*

$$\max_{j \in [\omega_k]} \{\Delta_{-K}(f^{a_j^k}(x_k))^\top s_k)\} \leq -\Gamma_1 |\theta(x_k)|, \tag{7.16}$$

$$\|s_k\|^2 \leq \Gamma_2 |\theta(x_k)|, \tag{7.17}$$

where $\Gamma_1 = 1$ and $\Gamma_2 = \frac{4}{T}$.

Proof: Since x_k is a non-critical point of ($\mathcal{VOP}_{a^k}(x_k)$), from Theorem 6.3, we have $s_k = s(x_k) \neq 0$ and $\theta(x_k) \neq 0$. Then, from assumption that $\nabla^2 f^i(x)$ is positive definite for all $i \in [p]$ and using Definition 7.1, we have

$$\begin{aligned}
& 0 \prec_K s_k^\top \nabla^2 f^i(x_k) s_k \text{ for all } i \in [p] \\
\implies & 0 \prec_K s_k^\top \nabla^2 f^{a_j^k}(x_k) s_k \text{ for all } j \in [\omega_k] \\
\implies & \nabla f^{a_j^k}(x_k)^\top s_k \prec_K \nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2} s_k^\top \nabla^2 f^{a_j^k}(x_k) s_k \text{ for all } j \in [\omega_k] \\
\implies & \Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k) < \Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2} s_k^2 \nabla^2 f^{a_j^k}(x_k) s_k) \text{ for all } j \in [\omega_k] \\
\implies & \max_{j \in [\omega_k]} \{\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k)\} < \max_{j \in [\omega_k]} \{\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2} s_k^2 \nabla^2 f^{a_j^k}(x_k) s_k)\} \\
& \qquad < \max_{j \in [\omega_k]} \{\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2} s_k^2 \nabla^2 f^{a_j^k}(x_k) s_k), \Delta_{-K}(\nabla f^{a_j^k} \\
& \qquad (x)^\top s_k)\} \overset{(7.1)}{=} \theta(x_k).
\end{aligned}$$

Now, since $\theta(x_k) < 0$ at non-critical point x_k, we have $\theta(x_k) = -|\theta(x_k)|$, which proves the first part (7.16).

For the second part (7.17), the Lagrangian of problem (7.4) is given by

$$L((\tau_1, s_k), (\lambda^1, \lambda^2)) = \tau_1 + \sum_{j=1}^{\omega_k} \lambda_j^1 (\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2} s_k^\top \nabla^2 f^{a_j^k}(x_k) s_k) - \tau_1)) + \sum_{j=1}^{\omega_k} \lambda_j^2$$
$$(\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k) - \tau_1)) + \lambda^3 \left(\frac{\|s_k\|^2}{2} - \frac{\Omega_k^2}{2} \right), \tag{7.18}$$

with $\lambda^1 = (\lambda_1^1, \lambda_2^1, \ldots, \lambda_{\omega_k}^1), \lambda^2 = (\lambda_1^2, \lambda_2^2, \ldots, \lambda_{\omega_k}^2) \in \mathbb{R}^m$ and $\lambda^3 \in \mathbb{R}$. The Karush-Kuhn-Tucker conditions for problem (7.4) are given by

$$\sum_{j=1}^{\omega_k} \lambda_j^1 \Delta_{-K}((\nabla f^{a_{j,r}^k}(x_k)^\top s_k)_{r \in [m]}) + (s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k) s_k)_{r \in [m]})) + \sum_{j=1}^{\omega_k} \lambda_j^2 \Delta_{-K}((\nabla f^{a_{j,r}^k}(x_k)^\top$$

$$s_k)_{r\in[m]}) + \lambda^3 s_k = 0, \tag{7.19}$$
$$\sum_{j=1}^{\omega_k} \lambda_j^1 = 1, \sum_{j=1}^{\omega_k} \lambda_j^2 = 1 \text{ and } \lambda_j^1, \lambda_j^2 \geq 0, j \in [\omega_k],$$
$$\Delta_{-K}(\nabla f^{a_{j,r}^k}(x_k)^\top s_k + \frac{1}{2} s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)s_k))_{r\in[m]}) - \tau_1 \leq 0, j \in [\omega_k],$$
$$\Delta_{-K}(\nabla f^{a_{j,r}^k}(x_k)^\top s_k) - \tau_1 \leq 0, j \in [\omega_k]$$
$$\lambda_j^1(\Delta_{-K}((\nabla f^{a_{j,r}^k}(x_k)^\top s_k + \frac{1}{2} s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)^\top s_k)_{r\in[m]}) - \tau_1) = 0, j \in [\omega_k],$$
$$\lambda_j^2(\Delta_{-K}((\nabla f^{a_{j,r}^k}(x_k)^\top s_k)_{r\in[m]}) - \tau_1) = 0, j \in [\omega_k],$$
$$\frac{\|s_k\|^2}{2} \leq \frac{\Omega_k^2}{2},\ \lambda^3 \left(\frac{\|s_k\|^2}{2} - \frac{\Omega_k^2}{2} \right) = 0,\ \lambda^3 \geq 0. \tag{7.20}$$

Here, we have two cases based on whether or not the trust-region constraint is active. When active, the constraint of trust-region in the Lagrangian function disappears. When inactive i.e. $\|s_k\| < \Omega_k$, the same result can be obtained by using complementary conditions, where the Lagrangian reduces to

$$L((\tau_1, s_k), (\lambda^1, \lambda^2)) = \tau_1 + \sum_{j=1}^{\omega_k} \lambda_j^1(\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2} s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k) - \tau_1) + \sum_{j=1}^{\omega_k} \lambda_j^2$$
$$(\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k) - \tau_1)) = \tau_1. \tag{7.21}$$

Next, from (7.19) and $\lambda^3 = 0$, we obtain

$$\sum_{j=1}^{\omega_k} \lambda_j^2 \Delta_{-K}((\nabla f^{a_{j,r}^k}(x_k)^\top s_k)_{r\in[m]}) = -\sum_{j=1}^{\omega_k} \lambda_j^1((\Delta_{-K}((\nabla f^{a_{j,r}^k}$$
$$(x_k)^\top s_k)_{r\in[m]} + (s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)s_k)_{r\in[m]}). \tag{7.22}$$

Then, using (7.21), (7.22), and property ((vii)) of (6.4.1), we have that

$$\begin{aligned} 2\tau_1 \quad &= \sum_{j=1}^{\omega_k} \lambda_j^1(\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k) + \frac{1}{2}(s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)s_k)_{r\in[m]}) \\ &- \sum_{j=1}^{\omega_k} \lambda_j^1(\Delta_{-K}(\nabla f^{a_{j,r}^k}(x_k)^\top s_k + s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)s_k)_{r\in[m]}) \\ &\leq \sum_{j=1}^{\omega_k} \lambda_j^1 \Delta_{-K}\left(\left(-\frac{1}{2} s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)s_k\right)_{r\in[m]}\right) = \frac{1}{2}\sum_{j=1}^{\omega_k} \lambda_j^1 \Delta_{-K}\left(\left(-s_k^\top \nabla^2 f^{a_{j,r}^k}(x_k)s_k\right)_{r\in[m]}\right). \end{aligned} \tag{7.23}$$

Now, from Assumption 7.5.3, we can have

$$
\begin{aligned}
&-(s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k)_{r\in[m]} \leq -M\|s_k\|^2\\
&\Delta_{-K}(-(s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k)_{r\in[m]}) \leq \Delta_{-K}(-M)\|s_k\|^2,
\end{aligned} \tag{7.24}
$$

and using (7.24), we obtain

$$
2\tau_1 \leq \frac{1}{2}\sum_{j=1}^{\omega_k} \lambda_j^1 \Delta_{-K}(-M)\|s_k\|^2 = \frac{1}{2}\Delta_{-K}(-M)\|s_k\|^2 < 0. \tag{7.25}
$$

Finally, taking $T = -\Delta_{-K}(-M)$, we have, from (7.25), that

$$
\begin{aligned}
&-4\tau_1 \geq -\frac{1}{2}\Delta_{-K}(-M)\|s_k\|^2 = \frac{1}{2}T\|s_k\|^2\\
&\Longrightarrow 4|\tau_1| \geq T\|s_k\|^2\\
&\Longrightarrow \|s_k\|^2 \leq \frac{4}{T}|\tau_1|.
\end{aligned}
$$

This proves (7.17). □

Next, we present an auxiliary lemma that will be used later to prove the convergence properties of the algorithms.

Lemma 7.2 *Suppose that Assumption 7.5.2 holds. Then, for each $j \in [\omega_k]$, we have*

$$
|-\Delta_{-K}(f^{a_j^k}(x_k+s_k) - f^{a_j^k}(x_k)) + \Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))| \leq \mathcal{O}(\|s_k\|^2).
$$

Proof: Using Taylor series expansion of the function $f^{a_j^k}$ around x_k, for each $j \in [\omega_k]$, we have

$$
\begin{aligned}
&|-\Delta_{-K}(f^{a_j^k}(x_k+s_k)) - f^{a_j^k}(x_k)) + \Delta_{-K}(-m_k^{a_j^k}(0) + m_k^{a_j^k}(s_k))|\\
=&\Bigg| -\Delta_{-K}((f^{a_j^k}(x_k) + \nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(\xi)s_k) - f^{a_j^k}(x_k)) + \Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k\\
&+\frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k)\Bigg|\\
=&\Bigg| -\Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(\xi)s_k) + \Delta_{-K}(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k)\Bigg|\\
\leq&\Bigg\|(\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k) - (\nabla f^{a_j^k}(x_k)^\top s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(\xi)s_k)\Bigg\|\\
=&\Bigg\|-\frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(\xi)s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k\Bigg\|,
\end{aligned} \tag{7.26}
$$

where $\xi = x_k + \tilde{\nu}s_k$ with $\tilde{\nu} \in (0,1)$, and s_k is the solution of Subproblem (7.4) at x_k.

Now, invoking Cauchy-Schwarz inequality and Assumption 7.5.2, we have the proof as

$$\begin{aligned}
& |-\Delta_{-K}(f^{a_j^k}(x_k+s_k)) - f^{a_j^k}(x_k)) + \Delta_{-K}(-m_k^{a_j^k}(0) + m_k^{a_j^k}(s_k))| \\
= \; & \frac{1}{2}\left(\|\nabla^2 f^{a_j^k}(\xi)\|\|s_k\|^2 + \|\nabla^2 f^{a_j^k}(x_k)\|\|s_k\|^2\right) \leq \mathcal{K}_1\|s_k\|^2 = \mathcal{O}\|s_k\|^2.
\end{aligned}$$

□

Next, using Lemma 7.1, we present a corollary that will be useful later for proving convergence.

Corollary 7.5.1 *Suppose that Assumption 7.5.2, Assumption 7.5.6, Theorem 6.4, Lemma 7.1 and Corollary 6.6.1 hold. Then, if x_k is a non-critical point and s_k is a solution of the subproblem 7.4, there exists a positive constant β such that, for all $j \in [\omega_k]$,*

$$\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k)) \geq \frac{\beta}{2}\frac{\Gamma_1|\theta(x_k)|}{\|s_k\|}\min\left\{\frac{\Gamma_1|\theta(x_k)|}{\|s_k\|\mathcal{K}_1}, \Omega_k\right\}, \tag{7.27}$$

with $\Gamma_1 = 1$. (7.27) does not hold when x_k is a critical point (for more details, see Chapter 6).

Proof: From (7.16) of Lemma 7.1, we have

$$\begin{aligned}
& \Delta_{-K}(f^{a_j^k}(x_k))^\top s_k) \leq -\Gamma_1|\theta(x_k)| \; \forall\; j \in [\omega_k] \\
\implies & -\Delta_{-K}(f^{a_j^k}(x_k))^\top s_k) \geq \Gamma_1|\theta(x_k)| \; \forall\; j \in [\omega_k].
\end{aligned} \tag{7.28}$$

Thus, using Corollary 6.6.1 and (7.28), we can obtain (7.27). □

Next, we present a theorem showing that the sequence of function values generated by the two algorithms, $F(x_{l(k)})$ for Max-NTRM or $\{(C_k^{i,r})_{r\in[m]}\}$ for Avg-NTRM, is monotonically decreasing.

Theorem 7.1 *Let $\{x_k\}$ be the sequence generated by Algorithm 4 (Max-NTRM) or Algorithm 5 (Avg-NTRM). Then, $\{(f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]}\}$ or $\{(C_k^{i,r})_{r\in[m]}\}$, respectively, for all $i \in [p]$, is a non-increasing sequence, i.e., for all k,*

$$(f^{i,r}(x_{l^{i,r}(k+1)}))_{r\in[m]} \preceq_K (f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]} \quad (\textit{for Algorithm 4}), \tag{7.29}$$

and

$$(f^{i,r}(x_{k+1}))_{r\in[m]} \preceq_K (C_{k+1}^{i,r})_{r\in[m]} \preceq_K (C_k^{i,r})_{r\in[m]} \quad (\textit{for Algorithm 5}). \tag{7.30}$$

Proof: We show the proof for the two algorithms separately. First, for Max-NTRM, simply from the definition of function $(f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]}$, we have

$$(f^{i,r}(x_k))_{r\in[m]} \preceq_K (f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]} \text{ for all } i \in [p]. \tag{7.31}$$

For Algorithm 4, the generated sequence $\{x_k\}$ contains only successful iterates. So, for x_k to be a successful iterate, we must have, from Step 8: of Algorithm 4, that

$$-\frac{\Delta_{-K}\left(f^{a_j^k}(x_k+s_k) - (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}\right)}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} \geq \eta_1 \geq 0 \text{ for all } j \in [\omega_k]. \tag{7.32}$$

Applying property (ii) of Lemma 6.4.1 to the inequality (7.32), we obtain, for the case when $a^k = a^{k'}$ with $k > k' \geq k - N_k$, that

$$f^{a_j^k}(x_{k+1}) \preceq_K \max_{0<q<N_k} (f^{a_j^k,r}(x_{k-q}))_{r\in[m]} = (f^{a_{j,r}^k}(x_{l^{j,r}(k)}))_{r\in[m]}, \tag{7.33}$$

for all $j \in [\omega_k]$. For the case when $a^k \neq a^{k'}$ for some $k > k' \geq k - N_k$, we again have from (7.31), for all $j \in [\omega_k]$, that

$$f^{a_j^k}(x_{k+1}) \preceq_k f^{a_j^k}(x_k) = (f^{a_j^k,r}(x_{l^{j,r}(k)}))_{r\in[m]}. \tag{7.34}$$

Thus, from (7.33), we have

$$\begin{aligned}
F(x_{l(k)}) &= \{(f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]}\}_{i\in[p]} \\
&\subseteq \{(f^{a_j^{l_j^r(k)}}(x_{l^{j,r}(k)}))_{r\in[m]}\}_{j\in[\omega_k]} + K \text{ by Proposition 2.1 in [18]} \\
&= \{(f^{a^k}(x_{l^{j,r}(k)}))_{r\in[m]}\}_{j\in[\omega_k]} + K \\
&\overset{(7.33)}{\subseteq} \{(f^{a_j^k,r}(x_{k+1}))_{r\in[m]}\}_{j\in[\omega_k]} + K \\
&\subseteq \{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} + K \\
&= F(x_{k+1}) + K.
\end{aligned} \tag{7.35}$$

Similarly, from (7.34), we have

$$\begin{aligned}
F(x_{l(k)}) &= \{(f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]}\}_{i\in[p]} \\
&\overset{(7.31)}{\subseteq} \{(f^{i,r}(x_k))_{r\in[m]}\}_{i\in[p]} + K \\
&\subseteq \{(f^{a_j^k,r}(x_k))_{r\in[m]}\}_{j\in[\omega_k]} + K \text{ by Proposition 2.1 in [18]} \\
&\subseteq \{(f^{a_j^k,r}(x_{k+1}))_{r\in[m]}\}_{j\in[\omega_k]} + K
\end{aligned}$$

$$
\begin{aligned}
&\subseteq \{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} + K \\
&= F(x_{k+1}) + K.
\end{aligned} \tag{7.36}
$$

Thus, combining (7.35) and (7.36), we have

$$
(f^{i,r}(x_{k+1}))_{r\in[m]} \preceq_K f^{i,r}(x_{l^{i,r}(k)})_{r\in[m]} \text{ for all } i \in [p]. \tag{7.37}
$$

Now, using (7.37) and the fact that $N_{k+1} = \min\{N_k + 1, \hat{N}\}$, we have, for all $i \in [p]$,

$$
\begin{aligned}
(f^{i,r}(x_{l^{i,r}(k+1)}))_{r\in[m]} &= \left(\max_{0\le b\le m(k+1)} f^{i,r}(x_{k+1-b})\right)_{r\in[m]} \\
&\preceq_K \left(\max_{0\le b\le m(k)+1} f^{i,r}(x_{k+1-b})\right)_{r\in[m]} \\
&= \left(\max_{0\le b-1\le m(k)} \{f^{i,r}(x_{k+1})), f^{i,r}(x_{k-(b-1)})\}\right)_{r\in[m]} \\
&= \left(\max\{f^{i,r}(x_{k+1})), \max_{0\le b\le m(k)} f^{i,r}(x_{k-b})\}\right)_{r\in[m]} \\
&= \left(\max\{f^{i,r}(x_{k+1})), f^{i,r}(x_{l^{i,r}(k)})\}\right)_{r\in[m]} \\
&\overset{(7.37)}{\preceq_K} (f^{i,r}(x_{l^{j,r}(k)}))_{r\in[m]}.
\end{aligned} \tag{7.38}
$$

Finally, using (7.38), we have

$$
F(x_{l(k)}) = \{(f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]}\}_{i\in[p]} \subseteq \{(f^{i,r}(x_{l^{i,r}(k+1)}))_{r\in[m]}\}_{i\in[p]} + K \subseteq F(x_{l(k+1)}) + K.
$$

Therefore, $F(x_{l(k+1)}) \preceq^l_K F(x_{l(k)})$ for all k and $\{F(x_{l(k)})\}$ is a non-increasing sequence.

Next, we consider Avg-NTRM and prove (7.30) for Algorithm 5. Note here that, for Max-NTRM, the values of $f^{a^k_j}(x_{l^j(k)})$ gets updated only in the case of successful iterate. Otherwise, it remain the same because of the max operation. However, for Avg-NTRM, the values of $C_k^{a^k_j}$ gets updated even when the current iterate is unsuccessful. Therefore, for Avg-NTRM, we first divide the sequence of iterations into two sets:

$$
I_1 = \{k : \rho^{a^k_j} \ge \eta_1, \forall j \in [\omega_k]\} \text{ and } I_2 = \{k : \exists l \in [\omega_k], \rho^{a^k_j} < \eta_1\}. \tag{7.39}
$$

I_1 represents the set of indices of successful iterations and I_2 the set of indices of unsuccessful iterations. Then, for $k \in I_1$, from the definition of $\rho_k^{a^k_j}$, Corollary 7.5.1, and the fact that $\rho_k^{a^k_j} \ge \eta_1$ for each $j \in [\omega_k]$, we have

$$
-\Delta_{-K}(f^{a^k_j}(x_k + s_k) - (C_k^{a^k_j,r})_{r\in[m]}) \ge \eta_1 \frac{\beta}{2} \frac{|\theta(x_k)|}{\|s_k\|} \min\left\{\frac{|\theta(x_k)|}{\|s_k\|\mathcal{K}_1}, \Omega_k\right\} > 0. \tag{7.40}
$$

Applying property (ii) of Lemma (6.4.1) to (7.40), we can write, for the case when $a^k = a^{k'}$ for all k' such that $k > k' \geq 0$, we have

$$(f^{a_j^k,r}(x_{k+1}))_{r\in[m]} \prec_K \frac{\mu_{k-1}q_{k-1}}{q_k}(C_{k-1}^{a_j^k,r})_{r\in[m]} + \frac{1}{q_k}(f^{a_{j,r}^k}(x_k))_{r\in[m]} = (C_k^{a_j^k,r})_{r\in[m]}, \quad (7.41)$$

and for the case when $a^k \neq a^{k'}$ for any k' such that $k > k' \geq 0$, we have

$$(f^{a_j^k,r}(x_{k+1}))_{r\in[m]} \prec_K (f^{a_j^k,r}(x_k))_{r\in[m]} = (C_k^{a_j^k,r})_{r\in[m]}. \quad (7.42)$$

Now, from both (7.41) and (7.42), we have

$$\begin{aligned}\{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]} &\subset \{(C_k^{a_j^k,r})_{r\in[m]}\}_{j\in[\omega_k]}\\ &\subset \{(f^{a_j^k,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} + K\\ &\subset F(x_{k+1}) + K \quad (7.43)\end{aligned}$$

Thus, from (7.43), we have, for all $i \in [p]$, that

$$(f^{i,r}(x_{k+1}))_{r\in[m]} \prec_K (C_k^{i,r})_{r\in[m]}. \quad (7.44)$$

Next, from the Definition 7.10 of $(C_k^{i,r})_{r\in[m]}$, for all $i \in [p]$, we can write

$$\begin{aligned}(C_k^{i,r})_{r\in[m]} - (C_{k+1}^{i,r})_{r\in[m]} &= (C_k^{i,r})_{r\in[m]} - \left(\frac{\mu_k q_k}{q_{k+1}}(C_k^{i,r})_{r\in[m]} + \frac{1}{q_{k+1}}(f^{i,r}(x_{k+1}))_{r\in[m]}\right)\\ &= \frac{1}{q_{k+1}}((C_k^{i,r})_{r\in[m]} - (f^{i,r}(x_{k+1}))_{r\in[m]}) \in \operatorname{int}(K) \quad (7.45)\end{aligned}$$

and

$$\begin{aligned}(C_{k+1}^{i,r})_{r\in[m]} - (f^{i,r}(x_{k+1}))_{r\in[m]} &= \left(\frac{\mu_k q_k}{q_{k+1}}(C_k^{i,r})_{r\in[m]} + \frac{1}{q_{k+1}}(f^{i,r}(x_{k+1}))_{r\in[m]}\right) \quad (7.46)\\ &\quad - (f^{i,r}(x_{k+1}))_{r\in[m]}\\ &= \frac{\mu_k q_k}{q_{k+1}}((C_k^{i,r})_{r\in[m]} - (f^{i,r}(x_{k+1}))_{r\in[m]}) \in \operatorname{int}(K). \quad (7.47)\end{aligned}$$

Combining (7.45) and (7.46), we get

$$F(x_{k+1}) = \{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} \prec_K^l \{(C_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} \prec_K^l \{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}. \quad (7.48)$$

Also note that if $\mu_k = 0$, from Definition 7.10 of $(C_k^{i,r})_{r\in[m]}$, we have the equality

$$F(x_{k+1}) = \{(f_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} = \{(C_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]}. \quad (7.49)$$

Therefore, for $\mu_k \geq 0$ and $q_k \geq 1$, first assembling (7.48) and (7.49), then applying property (viii) of Lemma 6.4.1, we can obtain

$$F(x_{k+1}) = \{(f_{k+1}^{i,r})_{r\in[m]})\}_{i\in[p]} \preceq_K^l \{(C_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} \preceq_K^l \{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}. \tag{7.50}$$

Next, we consider the unsuccessful case of $k \in I_2$. Here, since $x_{k+1} = x_k$, we have $(f_{k+1}^{a_j^k,r})_{r\in[m]} = (f_k^{a_j^k,r})_{r\in[m]}$, and $F_{k+1} = \{(f_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} = \{(f_k^{i,r})_{r\in[m]}\}_{i\in[p]} = F_k$. Now, to prove that (7.50) holds when $k \in I_2$, we have two subcases to consider: $k-1 \in I_1$ and $k-1 \in I_2$.

Case 1: For $k-1 \in I_1$, according to (7.48), we have $F(x_k) = \{(f_k^{i,r})_{r\in[m]})\}_{i\in[p]} \preceq_K^l \{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}$ for all $i \in [p]$. Then, from (7.45) and using $(f_{k+1}^{i,r})_{r\in[m]} = (f_k^{i,r})_{r\in[m]}$, we have, for all $i \in [p]$, that

$$\begin{aligned}
(f^{i,r}(x_{k+1}))_{r\in[m]} &= \frac{\mu_k q_k (f^{i,r}(x_{k+1}))_{r\in[m]} + (f^{i,r}(x_{k+1}))_{r\in[m]}}{q_{k+1}} \\
&= \frac{\mu_k q_k (f^{i,r}(x_k))_{r\in[m]} + (f^{i,r}(x_{k+1}))_{r\in[m]}}{q_{k+1}} \\
&\preceq_K \frac{\mu_k q_k (C_k^{i,r})_{r\in[m]} + (f^{i,r}(x_{k+1}))_{r\in[m]}}{q_{k+1}} \\
&= (C_{k+1}^{i,r})_{r\in[m]} \\
&= \frac{\mu_k q_k (C_k^{i,r})_{r\in[m]} + (f^{i,r}(x_{k+1}))_{r\in[m]}}{q_{k+1}} \\
&= \frac{\mu_k q_k (C_k^{i,r})_{r\in[m]} + (f^{i,r}(x_k))_{r\in[m]}}{q_{k+1}} \\
&\preceq_K \frac{\mu_k q_k (C_k^{i,r})_{r\in[m]} + (C_k^{i,r})_{r\in[m]}}{q_{k+1}} = (C_k^{i,r})_{r\in[m]}.
\end{aligned} \tag{7.51}$$

Therefore, we have $F(x_{k+1}) = \{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} \preceq \{(C_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} \preceq \{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}$.

Case 2: For $k-1 \in I_2$, suppose that $K = \{j' : 1 < j' \leq k, k-j' \in I_1\}$. If $K = \phi$, from the unsuccessful case of Step 8: in Algorithm 5, we have, for all $i \in [p]$ and for all $p' \in \{0, 1, 2, \ldots, k\}$, that

$$\begin{gathered}
(f^{i,r}(x_0))_{r\in[m]} = (f^{i,r}(x_{k-p'}))_{r\in[m]} = (f^{i,r}(x_{k+1}))_{r\in[m]}) \\
\text{because } (f^{a_j^k,r}(x_0))_{r\in[m]} = (f^{a_j^k,r}(x_{k-p'}))_{r\in[m]} = (f^{a_j^k,r}(x_{k+1}))_{r\in[m]}), \text{ for all } j \in [\omega_k].
\end{gathered}$$

As a consequence, from Definition 7.10 of $(C_k^{i,r})_{r\in[m]}$, we get, for all $i \in [p]$, that

$$(C_{k+1}^{i,r})_{r\in[m]} = (C_k^{i,r})_{r\in[m]} = (f_{k+1}^{i,r})_{r\in[m]}$$

$$\text{because } (C_{k+1}^{a_j^k,r})_{r\in[m]} = (C_k^{a_j^k,r})_{r\in[m]} = (f_{k+1}^{a_j^k,r})_{r\in[m]}, \text{ for all } j \in [\omega_k]. \tag{7.52}$$

Next, for $K \neq \phi$, setting $v = \min\{j' : j' \in K\}$, we get, for all $i \in [p]$, that

$$\begin{aligned}(f_{k-p'}^{i,r})_{r\in[m]} = (f_k^{i,r})_{r\in[m]} = (f_{k+1}^{i,r})_{r\in[m]} \text{ for all } p' \in \{0,1,2,\ldots,v-1\}\\ \text{as } (f_{k-p'}^{a_j^k,r})_{r\in[m]} = (f_k^{a_j^k,r})_{r\in[m]} = (f_{k+1}^{a_j^k,r})_{r\in[m]} \text{ for all } p' \in \{0,1,2,\ldots,v-1\}.\end{aligned} \tag{7.53}$$

From Definition 7.10 of $\{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}$, we have for all $i \in [p]$ that

$$\begin{aligned}&\mu_k q_k (C_k^{i,r})_{r\in[m]} + (f^{i,r}(x_{k+1}))_{r\in[m]}\\ &= \prod_{j=1}^{v-1} \mu_{k-j} q_{k-v+1} (C_{k-v+1}^{i,r})_{r\in[m]} + \sum_{i=0}^{v-2}\prod_{j=0}^{i} \mu_{k-j} (f_{k-i}^{i,r})_{r\in[m]} + (f_{k+1}^{i,r})_{r\in[m]}.\end{aligned} \tag{7.54}$$

Since $k-v \in I_1$, we must have, from (7.48), that $\{(f_{k-v+1}^{i,r})_{r\in[m]}\}_{i\in[p]} \preceq_K^l \{(C_{k-v+1}^{i,r})_{r\in[m]}\}_{i\in[p]}$. Next, using (7.52) and (7.54), we have, for all $i \in [p]$, that

$$q_{k+1}(f^{i,r}(x_{k+1}))_{r\in[m]} \tag{7.55}$$

$$\begin{aligned}&= (\prod_{j=0}^{v-1} \mu_{k-j} q_{k-v+1} + \sum_{i=0}^{v-2}\prod_{j=0}^{i} \mu_{k-j} + 1)(f_{k+1}^{i,r})_{r\in[m]}\\ &= (\prod_{j=0}^{v-1} \mu_{k-j} q_{k-v+1} (f_{k-v+1}^{i,r})_{r\in[m]} + \sum_{i=0}^{v-2}\prod_{j=0}^{i} \mu_{k-j} (f_{k-j}^{i,r})_{r\in[m]})\\ &\quad + (f^{i,r}(x_{k+1}))_{r\in[m]}\\ &\preceq_K \mu_k q_k (C_k^{i,r})_{r\in[m]} + (f^{i,r}(x_{k+1}))_{r\in[m]} = q_{k+1}(C_{k+1}^{i,r})_{r\in[m]}.\end{aligned} \tag{7.56}$$

Hence, we have $\{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} \preceq_K^l \{(C_{k+1}^{i,r})\}_{i\in[p]}$, and along the similar lines of (7.51), we have

$$\{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} \preceq_K^l \{(C_{k+1}^{i,r})\}_{i\in[p]} \preceq_k^l \{(C_{k+1}^{i,r})\}_{i\in[p]} \text{ for all } k \in I_2, i \in [p]. \tag{7.57}$$

When $\mu_k \neq 0$, from (7.57), we get

$$\{(f^{i,r}(x_{k+1}))_{r\in[m]}\}_{i\in[p]} \preceq_K^l \{(C_{k+1}^{i,r})\}_{i\in[p]} \preceq_k^l \{(C_{k+1}^{i,r})\}_{i\in[p]} \text{ for all } k \in I_2, i \in [p], \tag{7.58}$$

and when $\mu_k = 0$, from Definition 7.10 of $(C_k^{i,r})_{i\in[p]}$ and $k \in I_2$, we have, for all $i \in [p]$, that

$$(f_{k+1}^{i,r})_{r\in[m]} = (C_{k+1}^{i,r})_{r\in[m]} = (f_k^{i,r})_{r\in[m]}.$$

Therefore, from (7.58), we can conclude that $\{(f_k^{i,r})\}_{i\in[p]} \preceq_K^l \{(C_k^{i,r})\}_{i\in[p]}$ when $k-1 \in I_2$. Thus, (7.50) and (7.30) holds for all $k \in I_2$. This completes the proof. $\square$

Next, we present a corollary that ensures that, for a chosen a^k that satisfies (7.3), an accepted step for the objective function of ($\mathcal{VOP}_{a^k}(x_k)$) is also an accepted step for the objective functions $F_k = \{f_k^i\}_{i\in[p]}$ of ($\mathcal{SOP}_K^l$).

Corollary 7.5.2 *If a non-monotone trust region step s_k of Algorithm 4 or Algorithm 5 at x_k is accepted for the objective function $\widetilde{f}_k^{a^k} = (f_k^{a_1^k}, f_k^{a_2^k}, \ldots, f_k^{a_{\omega_k}^k})^\top$ of ($\mathcal{VOP}_{a^k}(x_k)$), then s_k is also accepted for the objective function F_k of ($\mathcal{SOP}_K^l$) at x_k.*

Proof: For Algorithm (4), using (7.35) and (7.36), we get

$$F(x_{k+1}) \preceq_K^l F(x_{l(k)}), \tag{7.59}$$

from which the conclusion follows for Algorithm 4. Similarly, for Algorithm 5, using (7.43), we get

$$F(x_{k+1}) \preceq_K^l \{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}, \tag{7.60}$$

from which the same conclusion follows for Algorithm 5. $\square$

The following theorem ensures that for a sequence $\{x_k\}$ generated by Algorithm 4 or Algorithm 5, number of unsuccessful steps is always finite and eventually we will definitely see a successful step.

Theorem 7.2 *Let $\{x_k\}$ be the sequence generated by Algorithm 4 or Algorithm 5. Suppose Assumption 7.5.1–7.5.6 hold true, and $|\theta(x_k)| > \epsilon > 0$. Then, for any k, there is a non-negative $p > 0$ such that x_{k+p} is a successful iterate.*

Proof: We prove this theorem by contradiction. Let us assume that there exists a k such that x_{k+p} is unsuccessful iterate for any arbitrary p. Then,

$$\exists\, l \in [\omega_k] \text{ such that } \rho_{k+p}^{a_l^{k+p}} < \eta_1, \text{ for } p = 0, 1, 2, \ldots, \tag{7.61}$$

Thus, from the unsuccessful case in Step 8: and Step 9: of Algorithm 4, we have

$$x_{k+p} = x_p,\ p = 0, 1, 2, \ldots \tag{7.62}$$

and

$$\lim_{p\to\infty} \Omega_{k+p} \to 0. \tag{7.63}$$

Next, from Lemma 7.1, intermediate step (4.11) in Chapter 6 to prove Corollary 7.5.1, and $|\theta(x)| > \epsilon$, we get

$$\|s_k\|^2 \leq \Gamma_2 |\theta(x_k)|, \tag{7.64}$$

and, for all $j \in [\omega_{k+p}]$,

$$\begin{aligned}
\Delta_{-K}(m_{k+p}^{a_j^{k+p}}(0) - m_{k+p}^{a_j^{k+p}}(s_{k+p})) &\geq \frac{\beta}{2} \frac{\Gamma_1 |\theta(x_{k+p})|}{\|s_{k+p}\|} \min\left\{ \frac{\Gamma_1 |\theta(x_{k+p})|}{\|s_{k+p}\| \mathcal{K}_1}, \Omega_{k+p} \right\} \\
&\geq \frac{\beta}{2} \frac{\Gamma_1 |\theta(x_{k+p})|}{\Gamma_2^{\frac{1}{2}} |\theta(x_{k+p})|^{\frac{1}{2}}} \min\left\{ \frac{\Gamma_1 |\theta(x_{k+p})|}{\Gamma_2^{\frac{1}{2}} |\theta(x_{k+p})|^{\frac{1}{2}} \mathcal{K}_1}, \Omega_{k+p} \right\} \\
&\geq \frac{\beta}{2} \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2} \min\left\{ \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2^{\frac{1}{2}} \mathcal{K}_1}, \Omega_{k+p} \right\}.
\end{aligned} \tag{7.65}$$

From Lemma 7.2 and (7.65), for all $j \in [\omega_{k+p}]$, we have that

$$\begin{aligned}
&\frac{-\Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(m_{k+p}^{a_j^{k+p}}(0) - m_{k+p}^{a_j^{k+p}}(s_{k+p}))} - 1 \\
&\leq \frac{-\Delta_{-K}(-m_{k+p}^{a_j^{k+p}}(s_{k+p})) - \Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(0 - m^{a_j^{k+p}}(s_{k+p}))} \\
&\leq \frac{\Delta_{-K}(m_{k+p}^{a_j^{k+p}}(s_{k+p})) - \Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(0) - \Delta_{-K}(m^{a_j^{k+p}}(s_{k+p}))} \\
&\leq \frac{\mathcal{O}(\|s_{k+p}\|)^2}{\frac{\beta}{2} \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2} \min\{ \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2, \mathcal{K}_1}, \Omega_k \}}.
\end{aligned} \tag{7.66}$$

For the same argument Lemma 7.2 and (7.65), for all $j \in [\omega_{k+p}]$, we also have that

$$\begin{aligned}
&\frac{\Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(m_{k+p}^{a_j^{k+p}}(0) - m_{k+p}^{a_j^{k+p}}(s_{k+p}))} + 1 \\
&\leq \frac{\Delta_{-K}(-m_{k+p}^{a_j^{k+p}}(s_{k+p})) + \Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(0 - m^{a_j^{k+p}}(s_{k+p}))} \\
&\leq \frac{\Delta_{-K}(-m_{k+p}^{a_j^{k+p}}(s_{k+p})) + \Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(0) - \Delta_{-K}(m^{a_j^{k+p}}(s_{k+p}))} \\
&= \frac{-\Delta_{-K}(-m_{k+p}^{a_j^{k+p}}(s_{k+p})) - \Delta_{-K}(f^{a_j^{k+p}}(x_{k+p} + s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(m^{a_j^{k+p}}(s_{k+p}))}
\end{aligned}$$

$$\leq \frac{\Delta_{-K}(\frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(\xi)s_k)}{\Delta_{-K}(m^{a_j^{k+p}}(s_{k+p}))}$$

$$= \frac{\Delta_{-K}(\frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(x_k)s_k + \frac{1}{2}s_k^\top \nabla^2 f^{a_j^k}(\xi)s_k)}{\Delta_{-K}(0) - \Delta_{-K}(m^{a_j^{k+p}}(s_{k+p}))} \leq \frac{\mathcal{O}(\|s_{k+p}\|)^2}{\frac{\beta}{2}\frac{\Gamma_1\epsilon^{\frac{1}{2}}}{\Gamma_2}\min\{\frac{\Gamma_1\epsilon^{\frac{1}{2}}}{\Gamma_2,\mathcal{K}_1}, \Omega_k\}}. \tag{7.67}$$

From (7.66) and (7.67), we have

$$\left|\frac{-\Delta_{-K}(f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(m_{k+p}^{a_j^{k+p}}(0) - m_{k+p}^{a_j^{k+p}}(s_{k+p}))} - 1\right| \leq \frac{\mathcal{O}(\|s_{k+p}\|)^2}{\frac{\beta}{2}\frac{\Gamma_1\epsilon^{\frac{1}{2}}}{\Gamma_2}\min\{\frac{\Gamma_1\epsilon^{\frac{1}{2}}}{\Gamma_2,\mathcal{K}_1}, \Omega_k\}}.$$

For sufficiently large p, from (7.63) and $\|s_{k+p}\| \leq \Omega_{k+p}$, we have, for all $j \in [\omega_{k+p}]$, that

$$\lim_{p\to\infty} \frac{-\Delta_{-K}(f^{a_j^{k+p}}(x_k+s_k) - f^{a_j^{k+p}}(x_{k+p}))}{\Delta_{-K}(0 - m_{k+p}^{a_j^{k+p}}(s_{k+p}))} = 1. \tag{7.68}$$

Here, considering only Algorithm 4, we recall the relation (7.31), to get, for all $j \in [\omega_{k+p}]$,

$$\begin{aligned}
&(f^{a_{j,r}^{k+p}}(x_{k+p}))_{r\in[m]} \preceq_K (f^{a_{j,r}^{k+p}}(x_{l_j^r(k+p)}))_{r\in[m]}\\
&\implies (f^{a_{j,r}^{k+p}}(x_{k+p}))_{r\in[m]} - (f^{a_{j,r}^{k+p}}(x_{l_j^r(k+p)}))_{r\in[m]} \in -K\\
&\implies \Delta_{-K}((f^{a_{j,r}^{k+p}}(x_{k+p}))_{r\in[m]} - (f^{a_{j,r}^{k+p}}(x_{l_j^r(k+p)}))_{r\in[m]}) \leq 0\\
&\overset{\text{Lemma 6.4.1 ((ii))}}{\implies} \Delta_{-K}((f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - (f^{a_{j,r}^{k+p}}(x_{l_j^r(k+p)}))_{r\in[m]})\\
&\qquad - (f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - (f^{a_{j,r}^{k+p}}(x_{k+p}))_{r\in[m]})) \leq 0\\
&\implies \Delta_{-K}(f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - (f^{a_{j,r}^{k+p}}(x_{l_j^r(k+p)}))_{r\in[m]})\\
&\overset{\text{Lemma 6.4.1 ((vii))}}{\leq} \Delta_{-K}((f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - (f^{a_{j,r}^{k+p}}(x_{k+p}))_{r\in[m]})).
\end{aligned} \tag{7.69}$$

From (7.69), we can conclude for all $j \in [\omega_{k+p}]$ that

$$\frac{-\Delta_{-K}(f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - (f^{a_{j,r}^{k+p}}(x_{k+p}))_{r\in[m]})}{\Delta_{-K}(0 - m_{k+p}^{a_j^{k+p}}(s_{k+p}))} \leq \frac{-(\Delta_{-K}(f^{a_j^{k+p}}(x_{k+p}+s_{k+p}) - (f^{a_{j,r}^{k+p}}(x_{l_j^r(k+p)}))_{r\in[m]}))}{\Delta_{-K}(0 - m_{k+p}^{a_j^{k+p}}(s_{k+p}))}. \tag{7.70}$$

Therefore, according to (7.68), (7.70) and $\eta_1 \in (0,1)$, when p is sufficiently large, we have $\rho_{k+p}^{a_j^{k+p}} \geq \eta_1$ and that contradicts (7.61). This concludes that for any k, there is a non-negative $p > 0$ such that x_{k+p} is successful iterate. Next, considering Algorithm 5, we recall the relation (7.30) to similarly have

$$(f^{a^{k+p}_{j,r}}(x_{k+p}))_{r\in[m]} \prec_K (C^{a^{k+p}_{j,r}}_{k+p})_{r\in[m]} \text{ for all } j \in [\omega_{k+p}]. \tag{7.71}$$

Then, it follows from (7.71) that, for all $j \in [\omega_{k+p}]$,

$$\Delta_{-K}(f^{a^{k+p}_{j,r}}(x_{k+p}+s_{k+p}) - (C^{a^{k+p}_{j,r}}_{k+p})_{r\in[m]}) \leq \ \Delta_{-K}(f^{a^{k+p}_{j}}(x_{k+p}+s_{k+p}) - (f^{a^{k+p}_{j,r}}(x_{k+p}))_{r\in[m]}). \tag{7.72}$$

From (7.72), we have, for all $j \in [\omega_{k+p}]$,

$$\frac{-\Delta_{-K}(f^{a^{k+p}_{j,r}}(x_{k+p}+s_{k+p}) - (f^{a^{k+p}_{j,r}}(x_{k+p}))_{r\in[m]})}{\Delta_{-K}(0 - m^{a^{k+p}_{j,r}}_{k+p}(s_{k+p}))} \tag{7.73}$$

$$\leq \ \frac{-\Delta_{-K}(f^{a^{k}_{j}}(x_{k+p}+s_{k+p}) - (C^{a^{k+p}_{j,r}}_{k})_{r\in[m]})}{\Delta_{-K}(0 - m^{a^{k+p}_{j}}(s_{k+p}))} \tag{7.74}$$

Therefore, according to (7.72), (7.73) and $\eta_1 \in (0,1)$, when p is sufficiently large we have $\rho^{a^{k+p}_{j}}_{k+p} \geq \eta_1$. This again contradicts (7.61) and we conclude that for any k, there is a non-negative $p > 0$ such that x_{k+p} is successful iterate. $\square$

In the following lemma, we show that the sequence $\{F(x_{l(k)})\}$ or $\{(C^{i,r}_k)_{r\in[m]}\}_{i\in[p]}$ converges to a limit as $k \to \infty$, and hence the sequence $\{x_k\}$ generated by Algorithm 4 or 5 is bounded.

Lemma 7.5.1 *Let $\{x_k\}$ be a sequence generated by Algorithm 4 or Algorithm 5. Suppose that Assumption 7.5.4 holds. Then $((f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]})$ or $((C^{i,r}_k))_{r\in[m]}$ converges to a limit when $k \to \infty$.*

Proof: From Assumption 7.5.4, the level set $\mathcal{L}_c$ of F is bounded. First we consider Algorithm 4. Here, by mathematical induction, we prove that if $x_{p'} \in \mathcal{L}_c$, then $x_{p'+1} \in \mathcal{L}_c$ for all $p' = 1, 2, \ldots, k$. For this, by Definition 7.7, we have $\{\left(f^{i,r}(x_{l^{i,r}(0)})\right)_{r\in[m]}\}_{i\in[p]} = F_0 = \{(f^{i,r}(x_0))_{r\in[m]}\}_{i\in[p]}$, for all $i \in [p]$. Additionally, since $\{x_k\}$ is a successful iterate, (7.37) also holds. Then, using these, and (7.29) of Theorem 7.1, we have, for all $i \in [p]$ and for all k, that

$$(f^{i,r}_{k+1})_{r\in[m]} \preceq_K \left(f^{i,r}(x_{l^{i,r}(k+1)})\right)_{r\in[m]} \preceq_K \left(f^{i,r}(x_{l^{i,r}(k)})\right)_{r\in[m]} \preceq_K f^i(x_0). \tag{7.75}$$

This can be further expressed as

$$F_{k+1} = \{(f^{i,r}_{k+1})_{r\in[m]}\}_{i\in[p]} \preceq^l_K \{(f^{i,r}(x_{l^{i,r}(k+1)}))_{r\in[m]}\}_{i\in[p]} \preceq^l_K \{(f^{i,r}(x_0)_{r\in[m]}\}_{i\in[p]}$$

$$= F(x_0) = F_0. \tag{7.76}$$

This shows that the sequence $\{x_k\}$ is contained in $\mathcal{L}_c$. Now, from Theorem 7.1 and (7.76), we have that $F(x_{l(k)})$, i.e., $\{(f^{i,r}(x_{l^{i,r}(k+1)}))_{r\in[m]}\}_{i\in[p]}$ is non-increasing sequence and bounded. Therefore, $\{F(x_{l(k)})\}$ admits a limit as $k \to \infty$.

Next, we consider Algorithm 5. Similar to above, by mathematical induction, we again prove that if $x_{p'} \in \mathcal{L}_c$, then $x_{p'+1} \in \mathcal{L}_c$ for all $p' = 1, 2, \ldots, k$. For this, by Definition 7.10, we have $\{(C_0^{i,r})_{r\in[m]}\}_{i\in[p]} = \{(f^{i,r}(x_0))_{r\in m}\}_{i\in[p]} = F_0$. Additionally, since $\{x_k\}$ is a successful iterate, (7.44) holds. Then, using these and (7.30) of Theorem 7.1, we have

$$F_{k+1} = \{(f_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} \preceq_K \{(C_{k+1}^{i,r})_{r\in[m]}\}_{i\in[p]} \preceq_K \{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]} \preceq_K \{f^i(x_0)\}_{i\in[p]} = F_0. \tag{7.77}$$

Here again the sequence $\{x_k\}$ is contained in $\mathcal{L}_c$. Since, from Theorem 7.1 and (7.77), we have that the sequence $\{(C_k^{i,r})_{r\in[m]}\}_{i\in[p]}$ is non-increasing and bounded, we conclude that $\{(C_k^{i,r})\}_{i\in[p]}$ admits a limit as $k \to \infty$. □

Finally, in the next theorem, we show that the sequence $\{x_k\}$ produced by the two algorithms converges to a critical point of ($\mathcal{SOP}_K^l$).

Theorem 7.3 *Let $\{x_k\}$ be a sequence of regular iterative points for F generated by Algorithm 4 or Algorithm 5. Suppose that Assumption 7.5.1-Assumption 7.5.6 hold. Then, we have*

(i) $\liminf_{k\to\infty}|\theta(x_k)| = 0$.

(ii) every limiting point of $\{x_k\}$ is a critical point for ($\mathcal{SOP}_K^l$).

Proof: First, we consider Algorithm 4. We can have two cases: (a) we have finite number of successful iterations followed by unsuccessful iterations for large k, (b) we have infinite number of successful iterations. In the first case, suppose k_0 is the index of the last successful iterate. At this point, if $|\theta(x_{k_0+1})|$ is still greater than ϵ, then this means that, from Lemma 7.2, we can find another successful iterate at an iteration with index larger than k_0. This is in direct contradiction with the statement of the first case and, thus, first case never holds. For the second case, we prove (i) by contradiction. Let us assume that there exists a constant $\epsilon > 0$ and $\mathcal{M} \subseteq \{0, 1, 2, \ldots\}$ such that for all $k \in \mathcal{M}$

$$|\theta(x_k)| > \epsilon. \tag{7.78}$$

First, we need to prove that

$$\lim_{k\to\infty} \Omega_k = 0. \tag{7.79}$$

For this, we consider the set of successful iterations I_1 and the set of unsuccessful iteration I_2 from (7.39). First, we show that (7.79) holds for $k \in I_1$, i.e.

$$\lim_{k\to\infty,\ k\in I_1} \Omega_k = 0. \tag{7.80}$$

From Lemma 7.2, it is clear that I_1 is an infinite set. Then, from Lemma 7.1, Lemma 7.5.1 and (7.78), we have

$$\Delta_{-K}(m^{a_j^k}(0) - m^{a_j^k}(s_k)) \geq \beta \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{2\Gamma_2^{\frac{1}{2}}} \min\left\{\frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2^{\frac{1}{2}} \mathcal{K}_1}, \Omega_k\right\}\ \forall\ j \in [\omega_k], \tag{7.81}$$

and, from the definition of $\rho_k^{a_j^k}$, (7.81) and Lemma 6.4.1 (vii), we have

$$\Delta_{-K}(f^{a_{j,r}^k}(x_{l_j^r(k)}))_{r\in[m]}) - \Delta_{-K}(f^{a_j^k}(x_k + s_k)) \geq \eta_1 \beta \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{2\Gamma_2^{\frac{1}{2}}} \min\left\{\frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2^{\frac{1}{2}} \mathcal{K}_1}, \Omega_k\right\} \tag{7.82}$$

$$\implies \Delta_{-K}(f^{a_j^k}(x_k + s_k)) \leq \Delta_{-K}(f^{a_{j,r}^k}(x_{l_j^r(k)}))_{r\in[m]}) - \eta_1 \beta \frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{2\Gamma_2^{\frac{1}{2}}} \min\left\{\frac{\Gamma_1 \epsilon^{\frac{1}{2}}}{\Gamma_2^{\frac{1}{2}} \mathcal{K}_1}, \Omega_k\right\}. \tag{7.83}$$

Then, applying Δ_{-K} to the inequality (7.37) and taking limit as $k \to \infty$ and $k \in I_1$ on both sides of (7.82), we obtain (7.79).

Next, for $k \in I_2$, if it is a finite set, it can be shown from (7.80) that (7.79) holds. Thus, I_2 is an infinite set. We define $\bar{K} = \{d_k : k = 1, 2, \ldots\}$ as a subset of I_2, where

$$d_1 = \min\{d' : d' \in I_2\},$$

and

$$d_{k+1} = \min\{d' \in I_2 : d' - 1 \in I_1, d' - 1 > d_k\},\ \forall\ k \geq 1.$$

For $k \geq 1$, according to the definition of d_k, we have $d_k - 1 \in I_1$. As per Case 2 of Step 9: of Algorithm 4, we have

$$\Omega_{d_k} \leq \gamma_2 \Omega_{d_k - 1}.$$

According to the definition of d_{k+1}, an h can be found which satisfies

$$d_k + h < d_{k+1} - 1 \text{ and } d_k + h \in I_2. \tag{7.84}$$

Let an integer h_k be the maximum that satisfies (7.84). Thus, as per Case 2 of Step 9: of Algorithm 4, we have

$$\Omega_{d_k+h_k+1} \leq \Omega_{d_k+h} \leq \Omega_{d_k} \leq \gamma_4 \Omega_{d_k-1}, \quad h = 0, 1, \ldots, h_k,$$

and consequently

$$\Omega_{d_k+h_k+1} \leq \Omega_{d_k+h} \leq \gamma_2 \Omega_{d_k-1}, \quad h = 0, 1, \ldots, h_k. \tag{7.85}$$

Finally, since $d_k + h_k + 1$ and $d_k - 1$ lie in I_1, from (7.80) and (7.85) we get that

$$\lim_{k\to\infty,\ k\in I_2} \Omega_k = 0. \tag{7.86}$$

Thus, combining (7.80) and (7.86), we get (7.79). Next, from Lemma 7.2 and (7.81), we have for all $j \in [\omega_k]$

$$\left| \frac{-\Delta_{-K}(f^{a_j^k}(x_k + s_k) - f^{a_j^k}(x_k))}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} - 1 \right| \leq \frac{\mathcal{O}(\|s_k\|)^2}{\frac{\beta}{2}\frac{\Gamma_1\epsilon^{\frac{1}{2}}}{\Gamma_2}\min\{\frac{\Gamma_1\epsilon^{\frac{1}{2}}}{\Gamma_2,\mathcal{K}_1}, \Omega_k\}}. \tag{7.87}$$

Then, from (7.79), (7.87) and $\|s_k\| \leq \Omega_k$, we have for all $j \in [\omega_k]$

$$\lim_{k\to\infty} \frac{-\Delta_{-K}(f^{a_j^k}(x_k + s_k) - f^{a_j^k}(x_k))}{\Delta_{-K}(m_k^{a_j^k}(0) - m_k^{a_j^k}(s_k))} = 1.$$

Further, from the definition of $\rho_k^{a_j^k}$ of Algorithm 4 and (7.31), we get, for all j,

$$\begin{aligned} &\frac{-\Delta_{-K}(f^{a_j^k}(x_k + s_k) - (f^{a_{j,r}^k}(x_k))_{r\in[m]})}{\Delta_{-K}(0 - m_k^{a_{j,r}^k}(s_k))} \\ \leq\ &\frac{-\Delta_{-K}(f^{a_j^k}(x_k + s_k) - (f^{a_{j,r}^k}(x_{l_j^r(k)})_{r\in[m]})}{\Delta_{-K}(0 - m_k^{a_j^k}(s_k))}. \end{aligned}$$

Therefore, for k sufficiently large, we have

$$\rho_k^{a_j^k} \geq \eta_1, j \in [\omega_k],$$

which shows that $\Omega_{k+1} \geq \Omega_k$ for sufficiently large k. This leads to the contradiction for (7.79), which shows that the assumption (7.78) is false and (i) is proved.

Next, for the second part of this theorem, from (i), we have

$$\lim_{k\to\infty} \inf |\theta(x_k)| = 0. \tag{7.88}$$

Since the sequence $\{x_k\}$ belongs to $\mathcal{L}_c$, which is bounded, $\{x_k\}$ is also bounded and has an accumulation point x^* with $\{x_k\}_{k\in\mathcal{M}}$ a subsequence of $\{x_k\}$ converging to x^*. From part (b) of Theorem 6.3, $\theta(x_k)$ is a continuous function on the set of regular point for $(\mathcal{SOP}^l_K)$. Thus, from (7.88), we have $\theta(x^*) = 0$. Hence, from part (b) of Theorem 6.3, x^* is a critical point for $(\mathcal{SOP}^l_K)$.

Next, for Algorithm 5, we can prove this same result by following similar approach as used in case of Algorithm 4. $\square$

7.6 Numerical Analysis.

In this section, we compare the performance of TRM, with our proposed methods Max-NTRM and Avg-NTRM by running experiments on 20 set optimization problems. These problems are defined in Appendix 12. For each problem, we randomly sample 100 initial points from a box region $\mathcal{S}$ of the form $[x_L, x_U]$, where $x_L = [L, L, \ldots, L]$, $x_U = [U, U, \ldots, U]$ and run the three algorithms from the same initial points. Since the initial points, and consequently the optimization problem, are constrained within the box $\mathcal{S}$, we need the next step to remain within the box as well i.e. $x_L < x_k + s_k < x_U$ x_k. Therefore, similar to [62], instead of solving the original subproblem (3.11), we solve the following:

$$\left.\begin{aligned}
&\min && t \\
&\text{subject to} && \Delta_{-K}\left(\nabla f^{a^k_j}(x_k)^\top s + \tfrac{1}{2} s^\top \nabla^2 f^{a^k_j}(x_k) s\right) - t \leq 0,\ j = 1, 2, \ldots, \omega_k, \\
& && \Delta_{-K}\left(\nabla f^{a^k_j}(x_k)^\top s\right) - t \leq 0,\ j = 1, 2, \ldots, \omega_k, \\
& && \|s\| \leq \Omega_k \\
& && x_L - x_k \leq s \leq x_U - x_k.
\end{aligned}\right\} \tag{7.89}$$

In order to compare the methods, we follow the performance profiling proposed by Dolan and More [43] for a set of solvers S (TRM, Max-NTRM, Avg-NTRM) on a set of Problems P (20 problems in our paper). The performance ratio $r_{p,s}$ of algorithm $s \in S$ for solving problem $p \in P$ is defined as:

$$r_{p,s} := \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}},$$

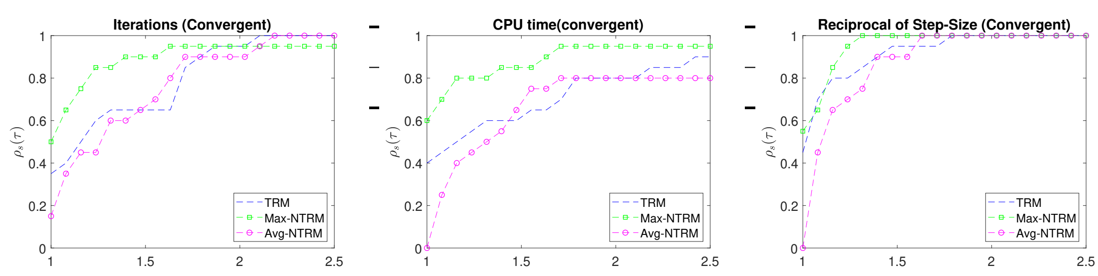
where $t_{p,s}$ is a metric of interest (e.g. number of successful convergence, number of iterations, CPU time, step size etc.) obtained by running algorithm s on problem p. For a problem p, the best performing (or winning) algorithm s will have $r_{p,s} = 1$, and the rest will have $r_{p,s} > 1$. Using $r_{p,s}$, we can obtain the performance profile ρ_s for algorithm s by computing the cummulative distribution function (CDF) of $r_{p,s}$, i.e.

$$\rho_s(\tau) := \frac{1}{|P|}|p \in P : r_{p,s} \leq \tau|.$$

In other words, $\rho_s(\tau)$ is the probability that the performance ratio $r_{p,s}$ of algorithm s lies in $[1, \tau]$. At $\tau = 1$, the value $\rho_s(1)$ gives the probability of algorithm s being the best performing algorithms, i.e. for how many problems, out of total $|P|$ problems, algorithms s was the winner. For example, $\rho_s(1) = 0.7$ means algorithm s is the best performing for 70 percent of the problems. However, for better judgement about the algorithms, we should also look at larger values of τ where better performing algorithms should have higher values of $\rho_s(\tau)$.

For running the experiments, we use Matlab on a system having Apple M2 chip with 8 CPU cores and 8 GB of RAM. For each problem, we run the algorithms for 100 initial points and compute $t_{p,s}$ over these points (either by taking mean over all 100 points or over a specific subset of points). All the parameters associated with different algorithms are given in Table 7.1. Maximum allowed number of iterations ($it_{\max}$) is set to 100 for all problems. A point for which an algorithms converges in less than $it_{\max}$ iterations is called a *convergent point* for that algorithm, otherwise its called a *non-convergent point*. For Avg-NTRM, $\mu_k = 0.5$ is fixed for all iterations k.

Table 7.1: Parameters of different algorithms used in all the experiments.

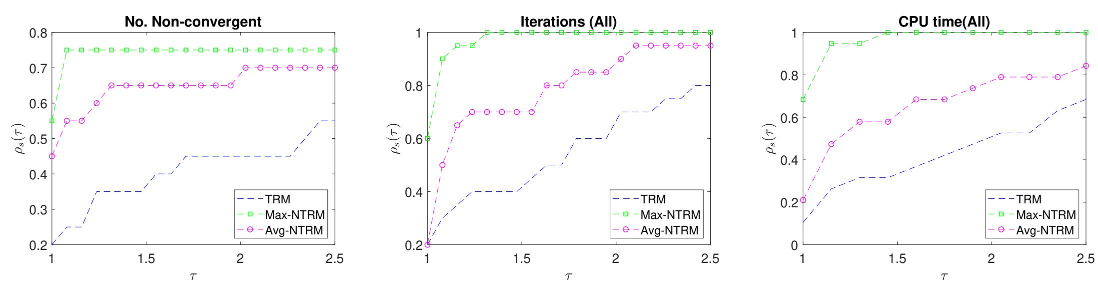


(a) No of iterations (b) CPU time (c) Step size

Figure 7.1: Performance profile of TRM, Max-NTRM and Avg-NTRM using three metrics: (a) no. of iterations to converge, (b) CPU time taken to converge, and (c) average step-size taken every iteration. For these plots, we consider only the case of conditioning on convergent point, i.e. the metrics are reported over only those initial points for which all three algorithms converged.

Table 7.2: Performance comparison of TRM, Max-NTRM (denoted as Max) and Avg-NTRM (denoted as Avg) for 20 set optimization problems using three metrics: (1) **No. Non-convergent**: no of initial points for which an algorithm failed to converge within 100 iterations, (2) **Iterations** (All): average number of iterations taken by an algorithm (averaged over all 100 initials points), (3) **Iterations** (Convergent): average number of iterations taken by an algorithm (averaged over only convergent points). For each metric, the best performing algorithm's value is marked in bold. Each problem is identified with a name, dimension n in argument space, and dimension m in image space. ↑ means higher values are better, and ↓ means lower values are better.

Type of SOP		**No. Non-convergent** ↓			**Iterations** (All) ↓			**Iterations** (Convergent) ↓		
Name	n, m	**TRM**	**Max**	**Avg**	**TRM**	**Max**	**Avg**	**TRM**	**Max**	**Avg**
ZDT1 [194]	$2, 2$	15	**10**	12	20.55	**17.78**	19.24	**6.38**	7.28	8.07
	$5, 2$	2	**0**	7	26.52	**11.2**	29.71	23.01	**11.27**	24.40
	$8, 2$	12	**0**	15	38.03	**18.82**	39.21	27.14	**16.0**	27.05
	$10, 2$	17	**0**	9	48.14	**24.05**	41.91	36.09	**21.38**	32.62
ZDT4 [194]	$10, 2$	**0**	**0**	**0**	6.34	**3.69**	5.79	6.34	**3.69**	5.79
DTLZ1 [37]	$6, 4$	33	**14**	**14**	41.86	26.04	**25.89**	12.18	**9.31**	9.36
DTLZ3 [37]	$5, 4$	45	38	**37**	**58.82**	64.7	64.93	4.53	4.12	**4.09**
DTLZ5 [37]	$3, 3$	49	**16**	32	51.8	**23.19**	37.44	3.55	**3.38**	5.77
	$5, 3$	30	**8**	**8**	33.3	16.82	**13.13**	**3.12**	5.09	3.88
	$7, 5$	28	**12**	15	29.85	**16.73**	20.19	**2.44**	2.89	4.95
Ex.4.3, [93]	$2, 4$	70	**2**	32	74.22	**25.48**	50.06	**13.96**	17.25	19.57
Hil [87]	$2, 2$	40	**33**	**33**	55.04	52.73	**51.23**	**15.5**	21.54	15.59
DGO1 [44]	$1, 2$	93	**91**	92	93.11	**91.21**	92.08	2.25	1.50	**1.37**
DGO2 [44]	$1, 2$	**2**	**2**	**2**	6.44	**5.36**	6.13	6.38	**5.31**	6.07
JOS1a [98]	$50, 2$	**26**	27	**26**	**30.75**	31.36	30.9	6.45	**5.97**	6.65
FDSa [56]	$2, 3$	**97**	**97**	**97**	**97.09**	97.22	97.12	**2.5**	9.0	4.0
Rosenbrock [176]	$4, 3$	79	27	**26**	82.18	**32.86**	33.63	15.78	9.94	**9.42**
Brown and Dennis [147]	$4, 5$	9	27	**0**	16.27	5.41	**5.12**	7.98	**4.36**	4.51
Trigonometric [147]	$4, 4$	5	**3**	**3**	5.11	**3.44**	3.46	**3.02**	3.17	3.18
Das and Dennis [36]	$5, 2$	**75**	76	78	**85.71**	87.21	88.56	34.87	**29.12**	38.12



(a) No. of non-convergence (b) Iterations (c) CPU time

Figure 7.2: Performance profile of TRM, Max-NTRM and Avg-NTRM using three metrics: (a) no of non-convergent points, (b) no. of iterations to converge, (c) CPU time taken to converge. For these plots, we report the metrics using all the initial points (convergent and non-convergent).

In terms of possibility of convergence, Figure 7.2(a) shows the performance profile of the three algorithms in terms of the probability of non-convergence (failure to converge)

Table 7.3: Performance comparison of TRM, Max-NTRM (denoted as Max) and Avg-NTRM (denoted as Avg) for 20 set optimization problems using three metrics: (1) **Step Size** (Convergent): average step size taken by an algorithms (averaged over only convergent points) (2) **CPU Time** (All): time taken by an algorithm (averaged over all 100 initials points), and (3) **CPU Time** (Convergent): time taken by an algorithm (averaged over only convergent points). For each metric, the best performing algorithm's value is marked in bold. Each problem is identified with a name, dimension n in argument space, and dimension m in image space. $\uparrow$ means higher values are better, and $\downarrow$ means lower values are better.

Type of SOP		**Step Size** (Convergent) $\uparrow$			**CPU Time** (All) $\downarrow$			**CPU Time** (Convergent) $\downarrow$		
Name	n, m	**TRM**	**Max**	**Avg**	**TRM**	**Max**	**Avg**	**TRM**	**Max**	**Avg**
ZDT1 [194]	2, 2	**0.401**	0.323	0.312	323.16	**264.27**	305.52	**88.44**	100.14	115.48
	5, 2	0.264	**0.284**	0.214	423.58	**150.20**	492.32	364.97	**151.22**	392.22
	8, 2	**0.282**	0.247	0.212	1773.97	**358.39**	1857.00	1304.90	**307.14**	1256.54
	10, 2	**0.239**	0.197	0.151	1363.05	**668.05**	1196.84	993.27	**595.37**	924.38
ZDT4 [194]	10, 2	0.490	**0.565**	0.526	780.97	**172.34**	737.94	780.96	**172.34**	737.94
DTLZ1 [37]	6, 4	0.413	**0.416**	0.391	1327.82	**782.96**	808.35	348.68	**270.14**	283.14
DTLZ3 [37]	5, 4	**0.507**	0.475	0.487	**2159.92**	2415.49	2615.04	153.01	**139.94**	146.90
DTLZ5 [37]	3, 3	**0.233**	0.214	0.215	817.94	**359.25**	538.37	**47.65**	52.15	71.65
	5, 3	0.282	**0.286**	0.281	918.04	366.48	**336.41**	**80.53**	107.29	96.43
	7, 5	**0.403**	0.368	0.372	2723.23	**721.12**	1741.04	217.89	**125.65**	400.02
Ex.4.3, [93]	2, 4	0.361	**0.634**	0.398	1343.72	**589.82**	887.89	**244.88**	401.05	345.05
Hil [87]	2, 2	**0.059**	0.053	0.057	629.40	692.98	**610.99**	**162.45**	261.92	174.51
DGO1 [44]	1, 2	0.322	**0.324**	0.323	1615.88	**897.09**	1740.80	33.76	**15.90**	27.18
DGO2 [44]	1, 2	0.741	**0.843**	0.773	44.90	**43.23**	46.96	44.89	**43.23**	46.96
JOS1a [98]	50, 2	1.773	**1.893**	1.879	**7710.54**	8824.40	7828.85	**1556.60**	1616.37	1603.77
FDSa [56]	2, 3	**0.059**	0.049	0.053	1237.73	1631.00	**1197.97**	**32.71**	130.31	48.13
Rosenbrock [176]	4, 3	0.237	**0.344**	0.260	1408.58	**568.57**	617.57	263.71	**170.67**	171.54
Brown and Dennis [147]	4, 5	1.998	**2.673**	2.617	424.84	146.96	**144.79**	203.20	**118.62**	129.64
Trigonometric [147]	4, 4	**0.155**	0.154	0.153	135.19	**88.63**	93.25	**81.27**	85.02	89.27
Das and Dennis [36]	5, 2	2.367	**3.019**	2.564	**1063.78**	1074.27	1152.04	420.49	**354.26**	482.39

within $it_{\max} = 100$ iterations. We observe that both non-monotone methods are better than TRM by a large margin, with Max-NTRM performing the best. The problem-wise values of the no. of non-convergent points used to plot the performance profile in Figure 7.2(a) are given in Table 7.2.

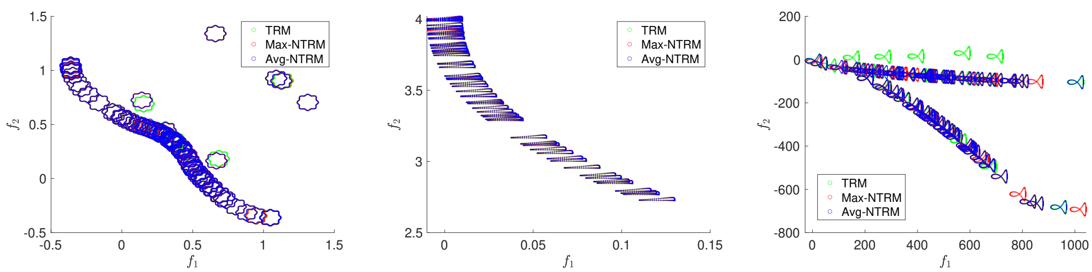
In Figure 7.1, we consider the case of conditioning on only the convergent points, so that we focus only on the aspect of speed of convergence. We look into speed of convergence with respect to three metrics: no of iterations in Figure 7.1(a), CPU time in Figure 7.1(b), and average step size in 7.1(c). We report these metrics over only those initial points for which all three algorithms converged. This case focuses purely on speed of convergence by conditioning on the event of convergent points, thus ignoring the effect of an algorithm not converging to the solution. The problem-wise values of the concerned metric are given in Table 7.2 and Table 7.3. In Figure 7.1(a), we see that TRM and Avg-NTRM are comparable to each other but Max-NTRM converges the fastest. It can also be seen that Avg-NTRM has more probability of convergence than TRM for $r_{p,s}$ less than equal to 1.75 (approx.). But, for $r_{p,s}$ being less than around 1.5, TRM has more probability than Avg-NTRM. For higher values of τ, TRM converges more than Avg-TRM. From Figure 7.1(b), we observe that TRM and Avg-NTRM are competitive but Max-NTRM takes less time than the other two methods.

In Figure 7.1(c), we show the performance profile based on the average size of the step taken by the three algorithms in an iteration. Compared to the metrics of no. of non-convergent points, iterations, and CPU time, for which lower values indicate better performance, step-size metric is opposite: higher values mean better performance. Since, for performance profiling in [43], the metric $t_{p,s}$ should have lower value for better performance, we take the $t_{p,s} = \frac{1}{\text{step-size}}$ as the performance metric. We compute the step size in terms of l_2 norm of the difference between the current and next iterate, i.e. step-size $= \|x_{k+1} - x_k\|$. The average step-size is calculated by first averaging over all iterations for each initial point and then averaging over the initial points. The problem-wise values of the average step-size used to plot the performance profile in Figure 7.1(c) are given in Table 7.3. From the figure, we see that TRM and Avg-NTRM are comparable to each other but Max-NTRM takes largest step-size among them. Therefore, from Figure 7.1(a), Figure 7.1(b), and Figure 7.1(c), we see that the three algorithms have a consistent behaviour, which confirm that Max-NTRM is the best performing algorithm with TRM and Avg-NTRM being comparable in terms of speed of convergence.

Next, in Figure 7.2, we consider the case when the metric is reported over all initial points (convergent and non-convergent). Here, for non-convergent points, we simply assign $it_{\max} = 100$ as the no. of iterations (as those many iterations were wasted by the algorithm on that point), and the total time spent on those $it_{\max} = 100$ iterations as the CPU time. These can be seen as penalty for failing to converge. Number of iterations is presented in Figure 7.2(b), where we see that Max-NTRM and Avg-NTRM both perform better than TRM by a large gap. Between Max-NTRM and Avg-NTRM, Max-NTRM converges faster by taking fewer iterations. CPU time is presented in Figure 7.2(c), where we find that both non-monotone schemes are far better than TRM, with Max-NTRM outperforming the others by a large margin.

Finally, if we look at the performance of the three algorithms for each problem, then, from Table 7.2, we observe that, in terms of possibility of convergence, non-monotone schemes are far better than TRM for all problems except 'Das and Dennis', where TRM is slightly better. On the other hand, if we focus purely on speed of convergence, then based on the metrics of Iterations (Convergent), Max-NTRM is better for ZDT1($n = 5, 8, 10, m = 2$), ZDT4, DTLZ5, DGO2, Brown and Dennis, Das and Dennis; Avg-NTRM is better for DGO1, and Rosenbrock; and, TRM is better for ZDT1 ($n = 2, m = 2$), DTLZ5 ($(n, m) = (5, 3), (7, 5)$), and FDSa. In terms of CPU time (Convergent), Max-NTRM is better for ZDT1 ($n = 5, 8, 10, m = 2$), ZDT4, DTLZ1, DTLZ5 ($n = 7, m = 5$). TRM is better for ZDT1 ($n = 2, m = 2$), DTLZ5 ($n = 3, m = 3$, $n = 5, m = 3$), Ex 4.3 [23], Hil, JOS1a, FDSa, Trigonometric. Finally, in terms of the step-size, from Table 7.3, we see that Max-NTRM and Avg-NTRM take larger steps than TRM for ZDT1 ($n = 5, m = 2$), ZDT4, DTLZ1, DTLZ5 ($n = 5, m = 3$), Ex 4.3, DGO2, JOS1a, Rosenbrock, Brown and Dennis, Das and Dennis, Hil, DGO1.

Figures 7.3 depicts the critical point frontiers for Hil, JOS1a, and DD1 problems. We observe that all the three algorithms converge to similar critical frontier. Overall, we



(a) Hil (b) JOS1 (c) DD1

Figure 7.3: Critical point frontiers obtained for Hil, JOS1, DD1 for TRM, Max-NTRM, and Avg-NTRM

find that NTRM is very beneficial for many examples. However, there can be problems e.g., DTLZ3, where TRM performs better. Therefore, Max-NTRM and Avg-NTRM is not a complete replacement for TRM, rather all three algorithms are important and they should be used as alternatives to each other for different kinds of problems.

7.7 Conclusion and future directions

In this article, we have studied two non-monotone trust-region methods, called for Max-NTRM and Avg-NTRM, to find critical point for ($\mathcal{SOP}_K^l$). For this, we have modified the trust-region scheme given in Chapter 6 based on non-monotone strategies of Ramiraz et al. [155] and Ghalavand et al. [61] from multi-objective optimization literature. The main modifications have been carried out in the step acceptance criteria by relaxing the strict monotonic decrement requirement of the monotone trust region scheme of Chapter 6. Instead of just considering the function values of current iteration as in the case of monotone TRM, we have taken into account the maximum over successive function values from last few iterations for Max-NTRM (Definition 7.7) and exponentially weighted moving average of successive function values upto current iteration for Avg-NTRM (Definition 7.10). These modifications have resulted in defining two new reduction ratios (Definition 7.6 and Definition 7.9) for Algorithm 4 and Algorithm 5, respectively. The well-definedness of these algorithms has been discussed in Subsection 7.4.1. Under some appropriate assumptions, the convergence analysis (Section 7.5) of two proposed methods have also been discussed. In convergence analysis of Algorithm 4 and Algorithm 5, we have derived the following

(i) At k-th iterate, the solution s_k of subproblem (7.4) satisfies certain relations that lower bounds the sufficient decrease of model $m_k^{a^k}$ (Corollary 7.5.1).

(ii) The sequence of maximum over previous successive function values $\{(f^{i,r}(x_{l^{i,r}(k)}))_{r\in[m]}\}_{i\in[p]}$ and weighted moving average of previous successive function values with current function values $\{(C_k^{i,r})_{r\in[m]})\}_{i\in[p]}$ decreases monotonically at every iteration and finally admit a limit as $k \to \infty$ (Lemma 7.5.1).

(iii) A step will be eventually accepted after only finite number of unsuccessful iterations along with a reduction in trust region radius (Theorem 7.2).

(iv) The global convergence of regular iterative sequence $\{x_k\}$ generated by Algorithm 4 or Algorithm 5 converges to a critical point for ($\mathcal{SOP}^l_K$) under bounded level set assumption (Theorem 7.3).

Finally, the numerical experiments performed for some well-known and newly defined 20 test problems exhibited the robustness, efficiency and advantages of the proposed non-monotone trust region scheme over monotone trust region method using the performance profile with respect to four performance metrics: probability of convergence, speed of convergence, execution time and average step-size.

****