A penalty method with trust-region mechanism for nonlinear bilevel optimization problem

B. El-Sobky*
bothinaelsobky@yahoo.com
Y. Abo-Elnaga †
yousrianaga@yahoo.com

November 11, 2017

Abstract

We present a penalty method with trust-region technique for nonlinear bilevel optimization problem in this paper. This method follows Dennis, El-Alem, and Williamson active set idea and penalty method to transform the nonlinear bilevel optimization problem to unconstrained optimization problem. This method maybe simpler than similar ideas and it does not need to compute a base of the null space. A trust-region technique is used to globalize the algorithm. Global convergence theorem is presented and applications to mathematical programs with equilibrium constraints are given.

Key Words: Nonlinear bilevel optimization problem; Active-set; Penalty method; Trust-region; Global convergence.

MSC 2010 : 90C30, 90B50, 65K05, 62C20.

1 Introduction

The nonlinear bilevel optimization (NBLO) problem, is a nested optimization problem which has two levels in hierarchy. The NBLO problem arises when two independent decision makers, DMs, ordered within a hierarchical structure, have conflicting objectives. The DM at the lower level (follower) has to optimize his objective under the given parameters from the upper level DM (leader), who, in return, with complete information on the possible reactions of the lower, selects the parameters so as to optimize his own objective. The DM with the upper level objective $f_u(x, y)$, takes the lead, and chooses his decision vector x. The DM with lower level objective $f_l(x, y)$, reacts accordingly by choosing his decision vector y to optimize his objective, parameterized in x. Note that the upper level decision maker is limited to influencing, rather than controlling, the lower levels outcome.

In this paper, we describe and analyze a penalty method with trust-region mechanism for the following NBLO problem

$$\begin{array}{ll} \min_{x,y} & f_u(x, y) \\ \textit{subject to} & g_u(x, y) \le 0, \\ \quad \min_y & f_l(x, y), \\ \quad \textit{subject to} & g_l(x, y) \le 0, \end{array} \tag{1.1}$$

*Department of Mathematics, Faculty of Science, Alexandria University, Alexandria, Egypt.
†Department of basic science, Higher Technological Institute, Tenth of Ramadan City , Egypt.

where $x \in \Re^{n_1}$, $y \in \Re^{n_2}$, $f_u : \Re^{n_1+n_2} \to \Re$, $f_l : \Re^{n_1+n_2} \to \Re$, $g_u : \Re^{n_1+n_2} \to \Re^{m_1}$, and $g_l : \Re^{n_1+n_2} \to \Re^{m_2}$.

There are several types of approaches suggested to solve NBLO problem, for example see [[1], [11], [19]]. The conventional solution approach to the NBLO problem is to transform the original two level problems into a single level one by replacing the lower level optimization problem with its Karush-Kuhn-Tucker (KKT) conditions [[1],[4],[20]]. This approach is used in this paper to transform NBLO problem to a general nonlinear programming (GNP) problem. This allows the use of the well-developed techniques for solving the GNP problems. The KKT conditions for the point y_* to be a local minimizer of the lower level problem (1.1) are

$$\begin{aligned} \nabla_y f_l(x,y) + \nabla_y g_l(x,y)\lambda &= 0, && (1.2)\\ g_l(x,y) &\leq 0, && (1.3)\\ (\lambda)_j g_{l_j}(x,y) &= 0, \quad j = 1, ..., m_2, && (1.4)\\ (\lambda)_j &\geq 0, \quad j = 1, ..., m_2, && (1.5) \end{aligned}$$

where $\lambda \in \Re^{m_2}$ is a multiplier vector associated with inequality constraint $g_l(x,y)$. For more details, see [9]. It follows that a necessary condition for (x, y, λ) to be an optimal solution of Problem (1.1), the point (y_*, λ_*) must satisfy conditions [1.2-1.5] at x_*. From this line of reasoning, Problem (1.1) is converted to the following GNP problem

$$\begin{array}{ll} \min_{x,y} & f_u(x,y) \\ \textit{subject to} & g_u(x,y) \leq 0, \\ & \nabla_y f_l(x,y) + \nabla_y g_l(x,y)\lambda = 0, \\ & g_l(x,y) \leq 0, \\ & \lambda_j g_{l_j}(x,y) = 0, \quad j = 1, ..., m_2, \\ & \lambda_j \geq 0, \quad j = 1, ..., m_2. \end{array} \qquad (1.6)$$

The above problem can be summarized as follows

$$\begin{array}{ll} \min_{\bar{x}} & f_u(\bar{x}) \\ \textit{subject to} & D_e(\bar{x}) = 0, \quad e \in E, \\ & D_i(\bar{x}) \leq 0, \quad i \in I, \end{array} \qquad (1.7)$$

where $\bar{x} = (x, y, \lambda)^T$, $E = \{1, ..., n_2 + m_2\}$, and $I = \{1, ..., m_1 + 2m_2\}$ such that $E \bigcap I = \emptyset$. We assume that $f_u(\bar{x})$, $D_e(\bar{x})$ for all $e \in E$, and $D_i(\bar{x})$ for all $i \in I$ are at least twice continuously differentiable functions.

Motivated by the active-set mechanism in [2], we define a 0-1 diagonal matrix $Z(\bar{x})$, whose diagonal entries are

$$z_i(\bar{x}) = \begin{cases} 1 & \text{if } i \in E, \\ 1 & \text{if } D_i(\bar{x}) \geq 0 \text{ for all } i \in I, \\ 0 & \text{if } D_i(\bar{x}) < 0 \text{ for all } i \in I. \end{cases} \qquad (1.8)$$

This matrix is used to convert Problem (1.7) to the following equality constrained optimization (ECO) problem

$$\begin{array}{ll} \textit{minimize} & f_u(\bar{x}) \\ \textit{subject to} & P(\bar{x})^T Z(\bar{x}) P(\bar{x}) = 0. \end{array}$$

For more details see [2].

Using a penalty method [[9], [10]] the above ECO problem transforms to the following unconstrained optimization problem.

$$\begin{array}{ll} \textit{minimize} & f_u(\bar{x}) + \frac{\rho}{2}\|Z(\bar{x})P(\bar{x})\|^2 \\ \textit{subject to} & \bar{x} \in \Re^{n_1+n_2+m_2}, \end{array} \qquad (1.9)$$

where $\rho \in \Re$ is a positive parameter.

The first-order necessary condition for the point $\bar{x}_*$ to be a local minimizer of Problem (1.9) is

$$\nabla_{\bar{x}} f_u(\bar{x}_*) + \rho \nabla P(\bar{x}_*) Z(\bar{x}_*) P(\bar{x}_*) \quad = \quad 0. \tag{1.10}$$

If the point $\bar{x}_*$ satisfies the first-order necessary conditions of Problem (1.7)[13], then it satisfies the first-order necessary conditions of Problem (1.9) but the converse is not necessarily true. So, we design our algorithm in such a way that, if the point $\bar{x}_*$ satisfies the first-order necessary condition of Problem (1.9), then it also satisfies the first-order necessary conditions of Problem (1.7).

As we know a trust-region mechanism can induce strongly global convergence, which are very important methods for solving unconstrained and constrained optimization problems and are more robust when they deal with rounding errors, so we used it in the proposed algorithm. One of the advantages of the trust-region mechanism is that it does not require the objective function of the model to be convex. However, in traditional trust-region mechanism, after solving a trust-region subproblem, we need to use some criteria to check if the trial step is acceptable. If not, the subproblem must be resolved with a reduced trust-region radius. For more details see [[6], [7], [17], [16], [21], [22], [24], [8]].

A trust-region quadratic subproblem associated with Problem (1.9) is

$$\begin{array}{ll} \textit{minimize} & q_k(s_k) = f_{u_k} + \nabla f_{u_k}^T s + \frac{1}{2} s^T H_k s + \frac{\rho_k}{2} \| Z_k (P_k + \nabla P_k^T s) \|^2 \\ \textit{subject to} & \|s\| \leq \delta_k, \end{array} \tag{1.11}$$

where $0 < \delta_k$ is a trust-region radius and H_k is the Hessian of $f_u(\bar{x_k})$ or an approximation to it.

The rest of this section introduces some notations. In Section 2 we introduce an outline of the proposed trust-region algorithm. Section 3 is devoted to analysis of the global convergence of the proposed algorithm. Section 4 contains implementation of the proposed algorithm and the results of test problems. Section 5 contains concluding remarks.

Subscripted functions denote function values at particular points; for example, $f_{u_k} = f_u(\bar{x_k})$, $P_k = P(\bar{x}_k)$, $\nabla f_{u_k} = \nabla f_u(\bar{x}_k)$, $\nabla P_k = \nabla P(\bar{x}_k)$, $Z_k = Z(\bar{x}_k)$ and so on. Sometimes, to simplify the notations, we need to rename the subsequence $\{k_i\}$ to $\{k\}$ to avoiding double indices. Finally, all norms are l_2-norms.

2 Description of the Trust-Region Algorithm

A detailed description of the trust-region algorithm for solving NBLO problem, is outlined in this section.

2.1 How to Compute s_k

In the proposed algorithm, we use a conjugate gradient method [18] to compute the trial step s_k. It is very cheap if the problem is large-scale and the Hessian is indefinite.

The steps for solving subproblem (1.11) are presented in the following algorithm

Algorithm 2.1 *: (A conjugate gradient algorithm to evaluate s_k)*
Step 1. *Set $0 = s_0 \in \Re^{n_1+n_2+m_2}$, $w_0 = -(\nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k)$, and $v_0 = w_0$.*
Step 2. *For $j = 1, ..., (n_1 + n_2 + m_2)$ do*

Compute $B_k = H_k + \rho_k \nabla P_k Z_k \nabla P_k^T$.
Compute $c_j = \frac{w_j^T w_j}{v_j^T B_k v_j}$.
Compute γ_j such that $\|s_j + \gamma_j v_j\| = \delta_k$.

If $v_j^T B_k v_j \leq 0$*, then set* $s_k = s_j + \gamma_j v_j$ *and Stop.*
Else, set $s_{j+1} = s_j + c_j v_j$ *and*
$w_{j+1} = w_j - c_j B_k v_j$.
If $\frac{w_{j+1}}{w_0} \leq \varepsilon_0$.
Set $s_k = s_{j+1}$ *and stop.*
Compute $\bar{q}_j = \frac{w_{j+1}^T w_{j+1}}{w_j^T w_j}$ *and the new direction is*
$v_{j+1} = w_{j+1} + \bar{q}_j v_j$.

To test the step s_k which is evaluated by using the above algorithm, to determine whether it will be accepted or not, we need to the following merit function

$$\ell(\bar{x}_k; \rho_k) = f_u(\bar{x}_k) + \frac{\rho_k}{2} \|Z(\bar{x}_k) P(\bar{x}_k)\|^2. \tag{2.12}$$

We define an actual reduction $Ared_k$ and a predicted reduction $Pred_k$ in the merit function to test whether the point $\bar{x}_{k+1} = \bar{x}_k + s_k$ will be taken as a next iterate or not.

$Ared_k$ in the merit function (2.12) is defined as follows

$$Ared_k = f_u(\bar{x}_k) - f_u(\bar{x}_{k+1}) + \frac{\rho_k}{2} [\| Z_k P_k \|^2 - \| Z_{k+1} P_{k+1} \|^2], \tag{2.13}$$

and $Pred_k$ is defined as

$$\begin{aligned} Pred_k &= q_k(0) - q_k(s_k) & (2.14) \\ &= -\nabla f_{u_k}^T s_k - \frac{1}{2} s_k^T H_k s_k + \frac{\rho_k}{2} [\| Z_k P_k \|^2 - \| Z_k (P_k + \nabla P_k^T s_k) \|^2]. & (2.15) \end{aligned}$$

2.2 How to test s_k and update δ_k

Our way of testing s_k and updating δ_k is presented in the following algorithm.

Algorithm 2.2 *: (Test* s_k *and update the trust-region radius algorithm)*
Choose $0 < \tau_1 < \tau_2 \leq 1$, $\delta_{max} > \delta_{min}$*, and* $0 < \eta_1 < 1 < \eta_2$.

While $\frac{Ared_k}{Pred_k} < \tau_1$*, or* $Pred_k \leq 0$.

Set $\delta_k = \eta_1 \| s_k \|$.
Evaluate a new s_k.

If $\tau_1 \leq \frac{Ared_k}{Pred_k} < \tau_2$*, then set* $\bar{x}_{k+1} = \bar{x}_k + s_k$.
$\delta_{k+1} = \max(\delta_k, \delta_{min})$.
End if.
If $\frac{Ared_k}{Pred_k} \geq \tau_2$*, then set* $\bar{x}_{k+1} = \bar{x}_k + s_k$.
$\delta_{k+1} = \min\{\delta_{max}, \max\{\delta_{min}, \eta_2 \delta_k\}\}$.
End if.

2.3 How to update ρ_k

To update the parameter $0 < \rho_k$, we use a scheme suggested by [23] and it is presented in the following algorithm.

Algorithm 2.3 *:(Updating the positive parameter* ρ_k *)*

Compute $Pred_k$ *given by (2.15).*

If

$$Pred_k \geq \| \nabla P_k Z_k P_k \| \min\{\| \nabla P_k Z_k P_k \|, \delta_k\}. \tag{2.16}$$

Set $\rho_{k+1} = \rho_k$.
Else, set $\rho_{k+1} = 2\rho_k$.
End if

Finally, the algorithm is stopped when either $\| \nabla f_{u_k} \| + \| \nabla P_k Z_k P_k \| \leq \epsilon_1$ or $\| s_k \| \leq \epsilon_2$ for some tolerances $0 < \epsilon_1$ and $0 < \epsilon_2$.

Main steps of our method is presented in the following algorithm.

Algorithm 2.4 *:(The trust-region algorithm)*
Step 0: *Given* $\bar{x}_0 \in \Re^{(n_1+n_2+m_2)}$. *Choose* $0 < \epsilon_1$, $0 < \epsilon_2$, τ_1, τ_2, η_1, *and* η_2, *such that* $0 < \tau_1 < \tau_2 \leq 1$ *and* $0 < \eta_1 < 1 < \eta_2$.
Choose δ_{min}, δ_{max}, *and* δ_0 *such that* $\delta_{min} \leq \delta_0 \leq \delta_{max}$.
Set $\rho_0 = 1$. *Set* $k = 0$.
Step 1: *If* $\| \nabla f_{u_k} \| + \| \nabla P_k Z_k P_k \| \leq \epsilon_1$, *then terminate the algorithm.*
Step 2: *Using Algorithm 2.1 to compute* s_k.
Step 3: *If* $\| s_k \| \leq \epsilon_2$, *then the algorithm stops.*
Step 4: *Set* $\bar{x}_{k+1} = \bar{x}_k + s_k$.
Step 5: *Compute* Z_{k+1} *given by (1.8).*
Step 6: *Test the step and update the trust-region radius using Algorithm (2.2).*
Step 7: *Update the positive parameter* ρ_k *using Algorithm (2.3).*
Step 8: *Set* $k = k + 1$ *and go to Step 1.*

The following section is devoted to global convergence analysis for the proposed algorithm.

3 Convergence Analysis

The following section is devoted to some assumptions under which the convergence theory is established.

3.1 Assumptions

Let $\{\bar{x}_k\}$ be the sequence of iterates generated by Algorithm (2.4) and let Ω be a convex subset of $\Re^{(n_1+n_2+m_2)}$ such that for all k, $\bar{x}_k$ and $\bar{x}_k + s_k$ are in Ω. On the set Ω, we assume:
[S_1]. The functions $f_u(\bar{x})$ and $P(\bar{x})$ are twice continuously differentiable functions for all $\bar{x} \in \Omega$.
[S_2]. All of $f_u(\bar{x})$, $\nabla f_u(\bar{x})$, $\nabla^2 f_u(\bar{x})$, $P(\bar{x})$, $\nabla P(\bar{x})$, are uniformly bounded in Ω.
[S_3]. The sequence of the Hessian matrices $\{H_k\}$ or an approximated to it is bounded.

In the above assumptions, we do not assume that $\nabla P(\bar{x})$ is linearly independent for all $x \in \Omega$. Then we may have other kinds of stationary points. They are presented in the following definitions.

Definition 3.1 *(A Fritz John point).*
A point $\bar{x}_* \in \Re$ *is called a Fritz John (FJ) point if there exist* $\tilde{\beta}_* \in \Re$ *and a Lagrange multiplier vector* $\mu_* \in \Re^(m_1 + 2m_2)$ *not all zero such that:*

$$\begin{aligned} \tilde{\beta}_* \nabla f_u(\bar{x}_*) + \nabla P(\bar{x}_*)\mu_* &= 0, & (3.1)\\ Z(\bar{x}_*)P(\bar{x}_*) &= 0, & (3.2)\\ (\mu_*)_i P_i(\bar{x}_*) &= 0, \quad i = 1, 2, ..., m_1 + 2m_2 & (3.3)\\ \tilde{\beta}_*,\ (\mu_*)_i &\geq 0, \quad i = 1, 2, ..., m_1 + 2m_2. & (3.4) \end{aligned}$$

conditions [3.1- 3.4] are called FJ conditions. For more details see [13].

If $\tilde{\beta}_* \neq 0$, then the FJ conditions are called the KKT conditions and the point $(\bar{x}_*, 1, \frac{\mu_*}{\tilde{\beta}_*})$ is called KKT point.

Definition 3.2 *(Infeasible FJ point)*
A point $\bar{x}_*$ is called an infeasible FJ point if there exist $\tilde{\beta}_* \in \Re$ and a Lagrange multiplier vector $\mu_* \in \Re^(m_1 + 2m_2)$ not all zero such that:

$$\begin{aligned} \tilde{\beta}_* \nabla f_u(\bar{x}_*) + \nabla P(\bar{x}_*)\mu_* &= 0, & (3.5)\\ \nabla P(\bar{x}_*) Z(\bar{x}_*) P(\bar{x}_*) &= 0 \quad but \ \|Z(\bar{x}_*)P(\bar{x}_*)\| > 0, & (3.6)\\ (\mu_*)_i P_i(\bar{x}_*) &= 0, \quad i = 1, 2, ..., m_1 + 2m_2 & (3.7)\\ \tilde{\beta}_*, \ (\mu_*)_i &\geq 0, \quad i = 1, 2, ..., m_1 + 2m_2. & (3.8) \end{aligned}$$

Conditions [3.5-3.8] are called infeasible FJ conditions.

If $\tilde{\beta}_* \neq 0$ then the infeasible FJ conditions are called an infeasible KKT conditions and the point $(\bar{x}_*, 1, \frac{\mu_*}{\tilde{\beta}_*})$ is called an infeasible KKT point.

The following two lemmas provide conditions equivalent to the conditions given in definitions (3.1-3.2).

Lemma 3.1 *Assume S_1-S_3. A subsequence $\{\bar{x}_{k_i}\}$ of the iteration sequence generated by Algorithm (2.4) satisfies infeasible FJ conditions if it satisfies:*
1) $\lim_{k_i \to \infty} \|Z_{k_i} P_{k_i}\| > 0$.
2) $\lim_{k_i \to \infty} \left\{ \min_s \left\{ \|Z_{k_i}(P_{k_i} + \nabla P_{k_i}^T s)\|^2 \right\} \right\} = \lim_{k_i \to \infty} \|Z_{k_i} P_{k_i}\|^2$.

Proof. Let $\bar{s}_k$ be the minimizer of $\min_s \|Z_k(P_k + \nabla P_k^T s)\|^2$, then it satisfies

$$\nabla P_k Z_k \nabla P_k^T \bar{s}_k + \nabla P_k Z_k P_k = 0. \tag{3.9}$$

If $\lim_{k \to \infty} \bar{s}_k = 0$, then $\lim_{k \to \infty} \{\nabla P_k Z_k P_k\} = 0$. Consider the limit in condition 2 which is equivalent to

$$\lim_{k \to \infty} \left\{ \bar{s}_k^T \nabla P_k Z_k \nabla P_k^T \bar{s}_k + 2\bar{s}_k^T \nabla P_k Z_k P_k \right\} = 0. \tag{3.10}$$

Multiplying (3.9) from the left $2\bar{s}_k^T$ and subtract it from (3.10), we have $\lim_{k \to \infty} \left\{ \bar{s}_k^T \nabla P_k Z_k \nabla P_k^T \bar{s}_k \right\} = 0$. This implies that $\lim_{k \to \infty} \{\nabla P_k Z_k P_k\} = 0$ if $\lim_{k \to \infty} \bar{s}_k \neq 0$. Hence in either case, we have

$$\lim_{k \to \infty} \{\nabla P_k Z_k P_k\} = 0.$$

Taking $(\mu_k)_i = (Z_k P_k)_i$, $i = 1, ..., m_1 + 2m_2$. Hence from Condition 1, we have $\lim_{k \to \infty} (\mu_k)_i \geq 0$, and $\lim_{k \to \infty} (\mu_k)_i > 0$, for some i. Then $\lim_{k \to \infty} \nabla P_k \mu_k = 0$ and hence the conditions of Definition (3.2) hold in the limit with $\tilde{\beta}_* = 0$.

Lemma 3.2 *Assume S_1-S_3. A subsequence $\{\bar{x}_{k_i}\}$ of the iteration sequence generated by Algorithm (2.4) satisfies FJ conditions if it satisfies:*
1) For all k_i, $\|Z_{k_i} P_{k_i}\| > 0$ and $\lim_{k_i \to \infty} Z_{k_i} P_{k_i} = 0$.
2) $\lim_{k_i \to \infty} \left\{ \min_s \left\{ \frac{\|Z_{k_i}(P_{k_i} + \nabla P_{k_i}^T s)\|^2}{\|Z_{k_i} P_{k_i}\|^2} \right\} \right\} = 1$.

Proof. Let $\bar{d}_k$ be a minimizer to the problem.

$$\min_{d \in \Re^n} \left\{ \|\hat{U}_k + Z_k \nabla P_k^T d\|^2 \right\}, \tag{3.11}$$

where $\hat{U}_k$ is a unit vector in the direction of Z_kP_k and $d = \frac{s}{\|Z_kP_k\|}$. Then $\bar{d}_k$ satisfies

$$\nabla P_k Z_k \nabla P_k^T \bar{d}_k + \nabla P_k Z_k \hat{U}_k = 0. \tag{3.12}$$

If $\lim_{k\to\infty} \bar{d}_k = 0$ in the above equation, then $\lim_{k\to\infty} \nabla P_k Z_k \hat{U}_k = 0$.

Consider the limit in Condition 2 which is equivalent to

$$\lim_{k\to\infty} \left\{ \min_{d\in\Re^n} \left\{ \|\hat{U}_k + Z_k \nabla P_k^T d\|^2 \right\} \right\} = 1.$$

Using the fact that $\bar{d}_k$ is a solution to the minimization Problem (3.11), we have

$$\lim_{k\to\infty} \{ \bar{d}_k{}^T \nabla P_k Z_k \nabla P_k^T \bar{d}_k + 2\hat{U}_k^T Z_k \nabla P_k^T \bar{d}_k \} = 0.$$

Multiplying (3.12) from the left by $2\bar{d}_k^T$ and subtract it from the above limit, we have

$$\lim_{k\to\infty} \bar{d}_k \nabla P_k Z_k \nabla P_k^T \bar{d}_k = 0.$$

This implies $\lim_{k\to\infty} \left\{ \nabla P_k Z_k \hat{U}_k \right\} = 0$ if $\lim_{k\to\infty} \bar{d}_k \neq 0$. Hence in both cases, we have

$$\lim_{k\to\infty} \left\{ \nabla P_k Z_k \hat{U}_k \right\} = 0.$$

The rest of the proof follows using arguments similar to those in the above lemma.

From the above lemma, we notice that, for any subsequence of the iteration sequence that satisfies FJ conditions, the corresponding subsequence of smallest singular values of $Z_k \nabla P_k^T$ is not bounded away from zero. That is the gradient of the active constraints are linear dependent.

3.2 Important lemmas

In this section, we introduce basic lemmas which are needed in the subsequent proofs.

Lemma 3.3 *Assume S_1 and S_3. Then $Z(x)P(x)$ is Lipschitz continuous in Ω.*

Proof. See Lemma (4.1) in [2].

From the above lemma, we conclude that $P(x)^T Z(x) P(x)$ is differentiable and $\nabla P(x) Z(x) P(x)$ is Lipschitz continuous in Ω.

Lemma 3.4 *At any iteration k, let $\hat{W}(\bar{x}_k)$ be a diagonal matrix whose diagonal entries are*

$$(\hat{w}_k)_i = \begin{cases} 1 & \text{if } (P_k)_i < 0 \text{ and } (P_{k+1})_i \geq 0, \\ -1 & \text{if } (P_k)_i \geq 0 \text{ and } (P_{k+1})_i < 0, \\ 0 & \text{otherwise}, \end{cases} \tag{3.13}$$

where $i = 1, 2, ..., m_1 + 2m_2$. Then

$$Z_{k+1} = Z_k + \hat{W}_k. \tag{3.14}$$

Proof. See Lemma (6.2) in [5].

Lemma 3.5 *Assume S_1 - S_3. At any iteration k, there exists a positive constant C_1 independent of k, such that*

$$\|\hat{W}_k P_k\| \leq C_1 \|s_k\|. \tag{3.15}$$

Proof. See Lemma (6.3) in [5].

The following lemma shows how accurate our definition of $Pred_k$ is as an approximation to $Ared_k$.

Lemma 3.6 *Assume S_1-S_3, then there exists a constant $C_2 > 0$ that does not depend on k, such that*

$$| \ Ared_k - Pred_k \ | \leq \kappa_2 \rho_k \|s_k\|^2. \tag{3.16}$$

Proof. From (2.13) and using (3.14) we have

$$Ared_k \quad = \quad f_u(\bar{x}_k) - f_u(\bar{x}_{k+1}) + \frac{\rho_k}{2}[P_k^T Z_k P_k - P_{k+1}^T(Z_k + \hat{W}_k)P_{k+1}].$$

From the above equation and (2.15), we have

$$\begin{aligned} | \ Ared_k - Pred_k \ | \quad \leq \quad & | \ f_u(\bar{x}_k) + \nabla f_u(\bar{x}_k)^T s_k + \frac{1}{2} s_k^T H_k s_k - f_u(\bar{x}_{k+1}) \ | \\ & + \frac{\rho_k}{2} \ | \ (P_k + \nabla P_k^T s_k)^T Z_k (P_k + \nabla P_k^T s_k) - P_{k+1}^T (Z_k + \hat{W}_k) P_{k+1} \ | \ . \end{aligned}$$

Using Cauchy-Schwarz inequality, then

$$\begin{aligned} |Ared_k - Pred_k| \quad \leq \quad & | \ \frac{1}{2} s_k^T (H_k - \nabla^2 f_u(\bar{x}_k + \xi_1 s_k)) s_k \ | \\ & + \rho_k \ | \ (\nabla P_k - \nabla P(\bar{x}_k + \xi_2 s_k)) Z_k P_k s_k \ | \\ & + \frac{\rho_k}{2} \ | \ s_k^T [\nabla P_k Z_k \nabla P_k^T - \nabla P(\bar{x}_k + \xi_2 s_k) Z_k \nabla P(\bar{x}_k + \xi_2 s_k)^T] s_k \ | \\ & + \frac{\rho_k}{2} \|\hat{W}_k P_k\|^2 + \rho_k \ | \ \nabla P(\bar{x}_k + \xi_2 s_k) \hat{W}_k P_k s_k \ | \\ & + \frac{\rho_k}{2} \ | \ s_k^T \nabla P(\bar{x}_k + \xi_2 s_k) \hat{W}_k \nabla P(\bar{x}_k + \xi_2 s_k)^T s_k \ |, \end{aligned}$$

for some ξ_1 and $\xi_2 \in (0, 1)$. Hence

$$\begin{aligned} |Ared_k - Pred_k| \quad \leq \quad & \frac{1}{2} \|H_k - \nabla^2 f_u(\bar{x}_k + \xi_1 s_k)\| \|s_k\|^2 \\ & + \rho_k \|\nabla P_k - \nabla P(\bar{x}_k + \xi_2 s_k)\| \|Z_k P_k\| \|s_k\| \\ & + \frac{\rho_k}{2} \|\nabla P_k Z_k \nabla P_k^T - \nabla P(\bar{x}_k + \xi_2 s_k) Z_k \nabla P(\bar{x}_k + \xi_2 s_k)^T\| \|s_k\|^2 \\ & + \frac{\rho_k}{2} \|\hat{W}_k P_k\|^2 + \rho_k \|\nabla P(\bar{x}_k + \xi_2 s_k)\| \|\hat{W}_k P_k\| \|s_k\| \\ & + \frac{\rho_k}{2} \|\nabla P(\bar{x}_k + \xi_2 s_k) \hat{W}_k \nabla P(\bar{x}_k + \xi_2 s_k)^T\| \|s_k\|^2. \end{aligned}$$

Using Inequality (3.15), then under the problem assumptions there exist positive constants a_1, a_2, and a_3 such that

$$| \ Ared_k - Pred_k \ | \leq a_1 \|s_k\|^2 + a_2 \rho_k \|s_k\|^3 + a_3 \rho_k \|s_k\|^2.$$

Since $\rho_k > 0$ then the above inequality can be written as

$$| \ Ared_k - Pred_k \ | \leq \kappa_2 \rho_k \|s_k\|^2.$$

This complete the proof.

Lemma 3.7 *Assume S_1-S_3. Then for all $k > \bar{k}$, there exists a positive constant C_3 independent of the iterates such that,*

$$Pred_k \geq C_3 \ \| \ \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \ \| \ \min\{\delta_k, \frac{\| \ \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \ \|}{\| \ B_k \ \|}\}. \tag{3.17}$$

Proof. Since the step s_k is obtained by approximating the solution of subproblem (1.11) using the conjugate gradient method, then it satisfies the fraction-of-Cauchy decrease condition

$$q_k(0) - q_k(s_k) \geq \varphi[q_k(0) - q_k(s_k^{cp})], \tag{3.18}$$

where $q_k(0)-q_k(s_k)$ and $q_k(0)-q_k(s_k^{cp})$ represent the predicted decrease obtained by s_k and the Cauchy step s_k^{cp} respectively. The Cauchy step s_k^{cp} is given by

$$s_k^{cp} = -\alpha_k^{cp}(\nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k), \tag{3.19}$$

such that the parameter α_k^{cp} is given by

$$\alpha_k^{cp} = \begin{cases} \frac{\|\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k\|^2}{\|B_k(\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k)\|^2} & \text{if } \frac{\|\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k\|^3}{\|B_k(\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k)\|^2} \leq \delta_k \\ & \text{and } \|B_k(\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k)\|^2 > 0, \\ \frac{\delta_k}{\|\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k\|} & \text{otherwise,} \end{cases} \tag{3.20}$$

where $B_k = H_k + r\nabla P(x_k)Z(x_k)\nabla P(x_k)^T$. Then we consider two cases:
i) If $s_k^{cp} = -\frac{\delta_k}{\|\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k\|}(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)$ and $\| \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \|^3 \geq \delta_k\|B_k(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)\|^2$, then

$$\begin{aligned} q_k(0) \quad - \quad & q_k(s_k^{cp}) = -(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)^T s_k^{cp} - \frac{1}{2} s_k^{cp^T} B_k s_k^{cp} \\ = \quad & \frac{\delta_k}{\| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \|} \| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \|^2 \\ - \quad & \frac{1}{2}\frac{\delta_k^2}{\| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \|^2}((\|B_k(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)\|^2) \\ \geq \quad & \frac{1}{2}\delta_k \| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \| . \end{aligned} \tag{3.21}$$

ii) If $s_k^{cp} = -\frac{\|\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k\|^2}{\|B_k(\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k)\|^2}(\nabla f_{u_k}+\rho_k\nabla P_k Z_k P_k)$, and $\| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \|^3 \leq \delta_k(\|B_k(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)\|^2)$, then

$$\begin{aligned} q_k(0) \quad - \quad & q_k(s_k^{cp}) = -(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)^T s_k^{cp} - \frac{1}{2} s_k^{cp^T} B_k s_k^{cp} \\ = \quad & \frac{1}{2}\frac{\| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \|^4}{\|B_k(\nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k)\|^2} \\ \geq \quad & \frac{\| \nabla f_{u_k} + \rho_k\nabla P_k Z_k P_k \|^2}{2 \| B_k \|}. \end{aligned} \tag{3.22}$$

From Inequalities (2.14), (3.18), (3.21), and (3.22), the proof completes.

In the following section, the convergence of the sequence of iterates is studied when ρ_k goes to infinity.

3.3 Convergence when ρ_k goes to infinity

From Algorithm 2.3, we notice that the sequence $\{\rho_k\}$ goes to infinity only when there exists an infinite subsequence of indices $\{k_j\}$ indexing iterates of acceptable steps that satisfy, for all $k \in \{k_j\}$

$$Pred_k < \|\nabla P_k Z_k P_k\| \min\{\|\nabla P_k Z_k P_k\|, \delta_k\}. \tag{3.23}$$

The following two lemmas show that a subsequence of the iteration sequence generated by the Algorithm (2.4) satisfies the FJ conditions or infeasible FJ conditions if $\rho_k \to \infty$ as $k \to \infty$.

Lemma 3.8 *Assume S_1-S_3 holds and $\rho_k \to \infty$ as $k \to \infty$. If there exists a subsequence $\{k_i\}$ of indices indexing iterates that satisfy $\|Z_k P_k\| \geq \varepsilon > 0$ for all $k \in \{k_i\}$, then a subsequence of the iteration sequence indexed $\{k_i\}$ satisfies the infeasible FJ conditions in the limit.*

Proof. The proof is by contradiction. Suppose that in the limit, there is no subsequence of iterates that satisfies the infeasible FJ conditions. Then, from Lemma (3.1) and Definition (3.2), we have $| \|Z_k P_k\|^2 - \|Z_k(P_k + \nabla Z_k^T s_k)\|^2 | \geq \varepsilon_1$ and $\|\nabla P_k Z_k P_k\| \geq \varepsilon_2$ respectively. Hence

$$\|\nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k\| \geq \rho_k \|\nabla P_k Z_k P_k\| - \|\nabla f_{u_k}\| \geq \rho_k \varepsilon_2 - \|\nabla f_{u_k}\| \geq \rho_k \varepsilon_2.$$

Since

$$\|B_k\| = \|H_k + \rho_k \nabla P_k Z_k \nabla P_k^T\| \leq \rho_k \|\nabla P_k Z_k \nabla P_k^T\| + \|H_k\|,$$

and using inequality (3.17), we have

$$Pred_k \geq C_3 \rho_k \varepsilon_2 \min\{\delta_k, \frac{\rho_k \varepsilon_2}{\rho_k \|\nabla P_k Z_k \nabla P_k^T\| + \|H_k\|}\}.$$

For k sufficiently large, we have

$$Pred_k \geq C_3 \rho_k \varepsilon_2 \min\{\delta_k, \frac{\varepsilon_2}{\|\nabla P_k Z_k \nabla P_k^T\|}\}. \tag{3.24}$$

Since $\rho_k \to \infty$, then there exists infinite number of acceptable iterates at which inequality (3.23) holds. From inequalities (3.23) and (3.24), we have

$$\|\nabla P_k Z_k P_k\| \min\{\|\nabla P_k Z_k P_k\|, \delta_k\} \geq C_3 \rho_k \varepsilon_2 \min\{\delta_k, \frac{\varepsilon_2}{\|\nabla P_k Z_k \nabla P_k^T\|}\}$$

Using assumptions S_2, the right hand side of the above inequality tends to infinity as $k \to \infty$, while the left hand side is bounded. This gives a contradiction unless $\rho_k \delta_k$ is bounded. That is $\delta_k \to 0$. Now, we consider two cases:
i) If $\|Z_k P_k\|^2 - \|Z_k(P_k + \nabla P_k^T s_k)\|^2 \geq \varepsilon$, then

$$\rho_k \{\|Z_k P_k\|^2 - \|Z_k(P_k + \nabla P_k^T s_k)\|^2\} \geq \rho_k \varepsilon \to \infty,$$

where $\rho_k \to \infty$, as $k \to \infty$. Hence $Pred_k \to \infty$ under assumptions S_2-S_3. That is, the left hand side of inequality (3.23) tends to infinity as $k \to \infty$, while the right hand side goes to zero because $\delta_k \to 0$. This gives a contradiction with the assumption in this case.
ii) If $\|Z_k P_k\|^2 - \|Z_k(P_k + \nabla P_k^T s_k)\|^2 \leq -\varepsilon$, then

$$\rho_k \{\|Z_k P_k\|^2 - \|Z_k(P_k + \nabla P_k^T s_k)\|^2\} \leq -\rho_k \varepsilon \to -\infty.$$

where $\rho_k \to \infty$ as $k \to \infty$. Similar to the first case, $Pred_k \to -\infty$. But $Pred_k > 0$ and this tends to a contradiction. These two contradictions prove the lemma.

The following lemma shows that the behavior of Algorithm (2.4) when $\lim inf_{k \to \infty} \|Z_k P_k\| = 0$ and $\rho_k \to \infty$ as $k \to \infty$.

Lemma 3.9 *Assume S_1-S_3 holds and $\rho_k \to \infty$ as $k \to \infty$. If there exists a subsequence $\{k_i\}$ of iterates that satisfies $\|Z_k P_k\| > 0$ for all $k \in \{k_i\}$ and $\lim_{k_i \to \infty} \|Z_{k_i} P_{k_i}\| = 0$, then a subsequence of the sequence of iterates indexed $\{k_i\}$ satisfies the FJ conditions in the limit.*

Proof. Suppose that there is no subsequence of the sequence of iterates that satisfies FJ conditions in the limit. Hence, from Lemma (3.2), we have

$$\frac{|\ \|Z_kP_k\|^2 - \|Z_k(P_k + \nabla P_k^T s_k)\|^2\ |}{\|Z_kP_k\|^2} \geq \varepsilon, \tag{3.25}$$

for all $0 < \varepsilon$ and k very large. We consider three cases:

i) If $\lim inf_{k\to\infty} \frac{s_k}{\|Z_kP_k\|} = 0$, then there is a contradiction with inequality (3.25).

ii) If $\lim sup_{k\to\infty} \frac{s_k}{\|Z_kP_k\|} = \infty$. From subproblem (1.11), we have

$$\nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k = -(B_k + \sigma_k I)s, \tag{3.26}$$

where $0 < \sigma_k$ is the Lagrange multiplier of the trust region constraint. Hence, we can write inequality (3.17) as follows

$$Pred_k \geq C_3 \parallel \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \parallel \min\{\delta_k, \frac{\parallel (B_k + \sigma_k I)s_k \parallel}{\parallel B_k \parallel}\}.$$

Since $B_k = H_k + \rho_k \nabla P_k Z_k \nabla P_k^T$, then

$$Pred_k \geq C_3 \parallel \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \parallel \min\{\delta_k, \frac{\parallel [\frac{1}{\rho_k} H_k + (\nabla P_k Z_k \nabla P_k^T + \frac{\sigma_k}{\rho_k} I)]s_k \parallel}{\parallel \frac{1}{\rho_k} H_k + \nabla P_k Z_k \nabla P_k^T \parallel}\}. \tag{3.27}$$

If $\rho_k \to \infty$, as $k \to \infty$, then there exists an infinite number of acceptable steps such that Inequality (3.23) holds. Since Inequality (3.23) can be written as

$$Pred_k < \|\nabla P_k\|^2 \|Z_kP_k\|^2, \tag{3.28}$$

then from inequalities (3.27) and (3.28), we can write

$$C_3 \parallel \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \parallel \min\{\delta_k, \frac{\parallel [\frac{1}{\rho_k} H_k + (\nabla P_k Z_k \nabla P_k^T + \frac{\sigma_k}{\rho_k} I)]s_k \parallel}{\parallel \frac{1}{\rho_k} H_k + \nabla P_k Z_k \nabla P_k^T \parallel}\} \leq Pred_k < a_4^2 \|Z_kP_k\|^2,$$

where $a_4 = sup_{x\in\Omega} \|\nabla P_k\|$. Dividing the above inequality by $\|Z_kP_k\|$, then

$$C_3 \parallel \nabla f_{u_k} + \rho_k \nabla P_k Z_k P_k \parallel \min\{\frac{\delta_k}{\|Z_kP_k\|}, \frac{\parallel [\frac{1}{\rho_k} H_k + (\nabla P_k Z_k \nabla P_k^T + \frac{\sigma_k}{\rho_k} I)]s_k \parallel}{\parallel \frac{1}{\rho_k} H_k + \nabla P_k Z_k \nabla P_k^T \parallel \|Z_kP_k\|}\} < a_4^2 \|Z_kP_k\|. \tag{3.29}$$

The right hand side of the above inequality goes to zero as $k \to \infty$. This implies that

$$\parallel \nabla f_{u_{k_i}} + \rho_{k_i} \nabla P_{k_i} Z_{k_i} P_{k_i} \parallel \frac{\parallel [\frac{1}{\rho_{k_i}} H_{k_i} + (\nabla P_{k_i} Z_{k_i} \nabla P_{k_i}^T + \frac{\sigma_{k_i}}{\rho_{k_i}} I)]s_{k_i} \parallel}{\parallel \frac{1}{\rho_{k_i}} H_{k_i} + \nabla P_{k_i} Z_{k_i} \nabla P_{k_i}^T \parallel \|Z_{k_i}P_{k_i}\|},$$

is bounded along the subsequence $\{k_i\}$ where $\lim_{k_i\to\infty} \frac{s_{k_i}}{\|Z_{k_i}P_{k_i}\|} = \infty$. That is, either $\frac{s_{k_i}}{\|Z_{k_i}P_{k_i}\|}$ lies in the null space of $\nabla P_{k_i} Z_{k_i} \nabla P_{k_i}^T + \frac{\sigma_{k_i}}{\rho_{k_i}} I$ or $\parallel \nabla f_{u_{k_i}} + \rho_{k_i} \nabla P_{k_i} Z_{k_i} P_{k_i} \parallel \to 0$. The first possibility occurs only when $\frac{\sigma_{k_i}}{\rho_{k_i}} \to 0$ as $k_i \to \infty$ and $\frac{s_{k_i}}{\|Z_{k_i}P_{k_i}\|}$ lies in the null space of the matrix $\nabla P_{k_i} Z_{k_i} \nabla P_{k_i}^T$ which contradicts assumption (3.25) and implies that a subsequence of the iteration sequence satisfies the FJ conditions in the limit. The second possibility implies $\parallel \nabla f_{u_{k_i}} + \rho_{k_i} \nabla P_{k_i} Z_{k_i} P_{k_i} \parallel \to 0$, as $k_i \to \infty$. Hence $\rho_{k_i} \|\nabla P_{k_i} Z_{k_i} P_{k_i}\|$ must be bounded and we have $\nabla f_{u_{k_i}} = 0$. This implies that a subsequence of the iteration sequence satisfies the FJ conditions in the limit.

iii) If $\lim sup_{k\to\infty}\frac{s_k}{\|Z_kP_k\|}<\infty$ and $\lim inf_{k\to\infty}\frac{s_k}{\|Z_kP_k\|}>0$. Therefore $\|s_k\|\to 0$. Hence, as in the second case, the right hand side of (3.29) goes to zero as $k\to\infty$. This implies that

$$\| \nabla f_{u_k}+\rho_k\nabla P_kZ_kP_k \| \frac{\| (\nabla P_kZ_k\nabla P_k^T+\frac{\sigma_k}{\rho_k}I)s_k \|}{\| \nabla P_kZ_k\nabla P_k^T \| \, \|Z_kP_k\|}\to 0.$$

This implies that, either $\| \nabla f_{u_k}+\rho_k\nabla P_kZ_kP_k \|\to 0$ or $\frac{\|(\nabla P_kZ_k\nabla P_k^T+\frac{\sigma_k}{\rho_k}I)s_k\|}{\|\nabla P_kZ_k\nabla P_k^T\|\|Z_kP_k\|}\to 0$. Similar case (ii), we can prove that a subsequence of the iteration sequence satisfies the FJ conditions in the limit. This completes the proof.

In the following section, the convergence of the sequence of iterates is studied when ρ_k is bounded.

3.4 Convergence when ρ_k is bounded

In this section we continue our analysis assuming that ρ_k is bounded. So, we assume that $\rho_k=\tilde{\rho}<\infty$ for all $k\geq\tilde{k}$.

Lemma 3.10 *Assume S_1-S_3. At any given iteration indexed k at which $\|\nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k\|+\|\nabla P_kZ_kP_k\|>\epsilon_1$, there exists a constant $C_4>0$ such that*

$$Pred_k\geq C_4\delta_k. \tag{3.30}$$

Proof. Since $B_k=H_k+\tilde{\rho}\nabla P_kZ_k\nabla P_k^T$, then under the problem assumptions we have $\|B_k\|\leq a_5$, where $0<a_5$ is a constant. Assume that $\|\nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k\|>\frac{\epsilon_1}{2}$, then $\|\nabla P_kZ_kP_k\|>\frac{\epsilon_1}{2}$. Using Lemma (3.7), we have

$$\begin{aligned} Pred_k &\geq C_3 \| \nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k \| \min\{\delta_k,\frac{\| \nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k \|}{\| B_k \|}\},\\ &\geq \frac{C_3\epsilon_1}{2}\min\{1,\frac{\epsilon_1}{2a_5\delta_{max}}\}\delta_k. \end{aligned}$$

Now, consider the case when $\|\nabla P_kZ_kP_k\|>\frac{\epsilon_1}{2}$. Using (2.16), we have

$$Pred_k\geq\frac{\epsilon_1}{2}\min\{\frac{\epsilon_1}{2\delta_{max}},1\}\delta_k.$$

Take $C_4=\min\{\frac{C_3\epsilon_1}{2}\min\{1,\frac{\epsilon_1}{2a_5\delta_{max}}\},\frac{\epsilon_1}{2}\min\{\frac{\epsilon_1}{2\delta_{max}},1\}\}$, the result follows.

Lemma 3.11 *Assume S_1-S_3. If $\|\nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k\|+\|\nabla P_kZ_kP_k\|>\epsilon_1$, then the condition $Ared_{k_j}\geq\tau_1Pred_{k_j}$ will be satisfied for some finite j, i.e., an acceptable step is found after finitely many trial steps calculations.*

Proof. Since $\|\nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k\|+\|\nabla P_kZ_kP_k\|>\epsilon_1$, then from Inequalities (3.16) and (3.30), we have

$$\mid\frac{Ared_k}{Pred_k}-1\mid=\frac{\mid Ared_k-Pred_k\mid}{Pred_k}\leq\frac{C_2\tilde{\rho}\delta_k^2}{C_4\delta_k}\leq\frac{C_2\tilde{\rho}\delta_k}{C_4}.$$

Now as the trial step s_{k_j} gets rejected, δ_{k_j} becomes small and eventually after finite number of trials, (i.e., for j finite), the acceptance rule will be met. This completes the proof.

Lemma 3.12 *Assume S_1-S_3. If $\|\nabla f_{u_k}+\tilde{\rho}\nabla P_kZ_kP_k\|+\|\nabla P_kZ_kP_k\|>\epsilon_1$, at a given iteration k and the j^{th} trial step satisfies*

$$\|s_{k^j}\|\leq\frac{(1-\tau_1)C_4}{2\tilde{\rho}C_2}, \tag{3.31}$$

then it must be accepted.

Proof. We prove this lemma by contradiction. Assume that the step s_{k^j} is rejected and inequality (3.31) holds. Then, from Inequalities (3.16)and (3.30) we have

$$(1-\tau_1) < \frac{\mid Ared_{k^j} - Pred_{k^j} \mid}{Pred_{k^j}} < \frac{C_2\tilde{\rho}\|s_{k^j}\|^2}{C_4\|s_{k^j}\|} \leq \frac{(1-\tau_1)}{2}.$$

This gives a contradiction and proves the lemma.

In the following theorem, we prove the main global convergence result for Algorithm (2.4).

Theorem 3.12 *Assume S_1-S_3. Then the sequence of iterates generated by the algorithm satisfies*

$$\liminf_{k\to\infty} \; [\ \|\nabla f_{u_k}\| + \|\nabla P_k Z_k P_k\|\ ] \;=\; 0. \tag{3.32}$$

Proof. In the first, we want prove that

$$\liminf_{k\to\infty}[\|\nabla f_{u_k} + \tilde{\rho}\nabla P_k Z_k P_k\| + \|\nabla P_k Z_k P_k\|] = 0. \tag{3.33}$$

We use a contradiction to prove 3.33. Suppose that $\|\nabla f_{u_k} + \tilde{\rho}\nabla P_k Z_k P_k\| + \|\nabla P_k Z_k P_k\| > \epsilon_1$ for all k. Let $k \geq \tilde{k}$ and a trial step indexed j of the iteration indexed k such that $k^j \geq \tilde{k}$. Using lemma (3.10), we have

$$\ell_{k^j} - \ell_{k^j+1} = Ared_{k^j} \geq \tau_1 Pred_{k^j} \geq \tau_1 C_4 \delta_{k^j}. \tag{3.34}$$

for any acceptable step indexed k^j. If $k \to \infty$, then

$$\lim_{k\to\infty} \delta_{k^j} = 0. \tag{3.35}$$

That is, the radius of the trust region is not bounded below.

Since $k^j > \bar{k}$, then at $j = 1$, we have $\delta_{k^1} \geq \delta_{min}$. Hence δ_{k^j} is bounded below in this case.

If $j > 1$, then there exists at least one rejected trial step. From lemma (3.12), we have for the rejected trial step,

$$\|s_{k^i}\| > \frac{(1-\tau_1)C_4}{2\tilde{\rho}C_2},$$

for all $i = 1, 2, ...j-1$. Since s_{k^i} is a rejected trial step, then from the way of updating the radius of trust region (see Step 5 algorithm (2.4)) and using the above inequality, we have

$$\delta_{k^j} = \eta_1 \|s_{k^{j-1}}\| > \eta_1 \frac{(1-\tau_1)C_4}{2\tilde{\rho}C_2}.$$

Hence δ_{k^j} is bounded below. But this contradicts (3.35). Therefore, the supposition is wrong. Hence,

$$\liminf_{k\to\infty}[\|\nabla f_{u_k} + \tilde{\rho}\nabla P_k Z_k P_k\| + \|\nabla P_k Z_k P_k\|] = 0.$$

But this also implies (3.32). This completes the proof of the theorem.

From the above theorem, we conclude that, given any ϵ_1, the algorithm terminates because $\|\nabla f_{u_k}\| + \|\nabla P_k Z_k P_k\| < \epsilon_1$.

4 Application

In this section, we introduce an extensive variety of possible numeric bi-level nonlinear programming problems to clarify the effectiveness of our Algorithm, since, Problem 1 [12] all the inner level functions are convex, Problem 2 [12], at fixed x, the inner problem is convex, Problem 3 [12], where the inner problem is quadratic in y, and there are no inner constraints, and Problems 4 and 5 [14] have quadratic functions in both levels. These problems are solved numerically with the help of Algorithm 2.4. Generally speaking, all problems are solved in a small number of iterations and the all solutions of these problems by our algorithm compared with the optimal solutions from corresponding references are introduced in Table 1. It is clear from the results that our approach is capable for treating nonlinear bilevel programming problems even the upper and the lower levels are convex or not and the computed results converge to the optimal solution, also the solution obtained are the same to the optimal that reported in literature.

Problem 1:

$$\begin{array}{ll} \min_x & f_u = 16x^2 + 9y^2 \\ \textit{subject to} & -4x + y \leq 0, \\ & x \geq 0, \\ \quad \min_y & f_l = (x + y - 20)^4, \\ \quad \textit{subject to} & 4x + y - 50 \leq 0, \\ & y \geq 0. \end{array}$$

We transform the above problem to the form (1.7) by introduce KKT conditions of the lower problem, which reads

$$\begin{array}{ll} \min_{x,y} & f_u = 16x^2 + 9y^2 \\ \textit{subject to} & 4(x + y - 20)^3 + \lambda_1 - \lambda_2 = 0, \\ & \lambda_1(4x + y - 50) = 0, \\ & \lambda_2 y = 0, \\ & 4x + y - 50 \leq 0, \\ & -4x + y \leq 0, \\ & (x, y)^T \geq 0, \quad (\lambda_1, \lambda_2)^T \geq 0. \end{array}$$

Applying Algorithm 2.4, we have the solution $(x, y)^T = (11.25, 5)$, $f_u = 2250$, and $f_l = 197.753$

Problem 2:

$$\begin{array}{ll} \min_x & f_u = x^3 y_1 + y_2 \\ \textit{subject to} & 0 \leq x \leq 1, \\ \min_y & f_l = -y_2 \\ \textit{subject to} & x y_1 \leq 10, \\ & y_1^2 + x y_2 \leq 1, \\ & y_2 \geq 0. \end{array}$$

We transform the above problem to the form (1.7) by introduce KKT conditions of the lower problem, which reads

$$\begin{array}{ll} \min_{x,y} & f_u = x^3 y_1 + y_2 \\ \textit{subject to} & \lambda_1 x + 2\lambda_2 y_1 = 0, \\ & -1 + \lambda_2 x - \lambda_3 = 0, \\ & \lambda_1 (x y_1 - 10) = 0, \\ & \lambda_2 (y_1^2 + x y_2 - 1) = 0, \\ & \lambda_3 y_2 = 0, \\ & x y_1 \leq 10, \\ & y_1^2 + x y_2 \leq 1, \\ & 0 \leq x \leq 1, \\ & y_2 \geq 0, \quad (\lambda_1, \lambda_2, \lambda_3)^T \geq 0. \end{array}$$

Applying Algorithm 2.4, we have the solution $(x, y_1, y_2)^T = (1, 0, 1)$, $f_u = 1$, and $f_l = 0$.

Test problem 3:

$$\begin{array}{ll} \min_{x} & f_u = (x-3)^2 + (y-2)^2 \\ \textit{subject to} & -2x + y - 1 \leq 0, \\ & x - 2y + 2 \geq 0, \\ & x + 2y - 14 \geq 0, \\ & 0 \leq x \leq 8, \\ \min_{y} & f_l = (y-5)^2. \end{array}$$

We transform the above problem to the form (1.7) by introduce KKT conditions of the lower problem, which reads

$$\begin{array}{ll} \min_{x} & f_u = (x-3)^2 + (y-2)^2 \\ \textit{subject to} & 2(y-5) = 0, \\ & -2x + y - 1 \leq 0, \\ & x - 2y + 2 \geq 0, \\ & x + 2y - 14 \geq 0, \\ & 0 \leq x \leq 8. \end{array}$$

Applying Algorithm 2.4, we have the solution $(x, y)^T = (3, 5)$, $f_u = 9$, and $f_l = 0$.

Problem 4:

$$\begin{array}{ll} \min_{x} & f_u = y_1^2 + y_2^2 + x^2 - 4x \\ \textit{subject to} & 0 \leq x \leq 2, \\ \quad \min_{y_1, y_2} & f_l = y_1^2 + \frac{1}{2} y_2^2 + y_1 y_2 + (1-3x) y_1 + (1+x) y_2 \\ \quad \textit{subject to} & 2y_1 + y_2 - 2x \leq 1, \\ & y_1 \geq 0, \ \textit{and} \ y_2 \geq 0. \end{array}$$

We transform the above problem to the form (1.7) by introduce KKT conditions of the lower problem, which reads

$$\begin{array}{ll} \min_{x} & f_u = y_1^2 + y_2^2 + x^2 - 4x \\ \textit{subject to} & 2y_1 + y_2 + (1-3x) + 2\lambda_1 - \lambda_2 = 0, \\ & y_2 + y_1 + (1+x) + \lambda_1 - \lambda_3 = 0, \\ & \lambda_1 (2y_1 + y_2 - 2x - 1) = 0, \\ & \lambda_2 y_1 = 0, \quad \lambda_3 y_2 = 0, \\ & 2y_1 + y_2 - 2x \leq 1, \\ & 0 \leq x \leq 2, \\ & (y_1, y_2)^T \geq 0, \quad (\lambda_1, \lambda_2, \lambda_3)^T \geq 0. \end{array}$$

Applying Algorithm 2.4, the solution of the above example is $(x, y_1, y_2)^T = (0.8461, 0.7692, 0)$, $f_u = -2.0769$, and $f_l = -0.5917$.

Problem Name	Solution corresponding to Algorithm 2.4	Optimal solution from corresponding references
Problem 1	$(x, y)^T = (11.25, 5)$	$(x, y)^T = (11.25, 5)$
Problem 2	$(x, y_1, y_2)^T = (1, 0, 1)$	$(x, y_1, y_2)^T = (1, 0, 1)$
Problem 3	$(x, y)^T = (3, 5)$	$(x, y)^T = (3, 5)$
Problem 4	$(y_1, y_2, x)^T = (0.7692, 0, .8461)$	$(y_1, y_2, x)^T = (0.7692, 0, .8461)$
Problem 5	$(x_1, x_2, y_1, y_2, y_3)^T = (0.609, 0.391, 0, 0, 1.828)$	$(x_1, x_2, y_1, y_2, y_3)^T = (0.609, 0.391, 0, 0, 1.828)$

Table 1: The comparison of the optimal solution

Problem 5:

$$\begin{array}{ll} \min_x & f_u = y_1^2 + y_3^2 - y_1y_3 - 4y_2 - 7x_1 + 4x_2 \\ \textit{subject to} & x_1 + x_2 \leq 1, \\ & (x_1, x_2)^T \geq 0 \\ \min_{y_1,y_2} & f_l = y_1^2 + \frac{1}{2}y_2^2 + \frac{1}{2}y_3^2 + y_1y_2 + (1 - 3x_1)y_1 + (1 + x_2)y_2 \\ \textit{subject to} & 2y_1 + y_2 - y_3 + x_1 - 2x_2 + 2 \leq 0, \\ & (y_1, y_2, y_3)^T \geq 0. \end{array}$$

$$\begin{array}{ll} \min_x & f_u = y_1^2 + y_3^2 - y_1y_3 - 4y_2 - 7x_1 + 4x_2 \\ \textit{subject to} & 2y_1 + y_2 + (1 - 3x_1) + 2\lambda_1 - \lambda_2 = 0, \\ & y_2 + y_1 + (1 + x_2) + \lambda_1 - \lambda_3 = 0, \\ & \lambda_1(2y_1 + y_2 - y_3 + x_1 - 2x_2 + 2) = 0, \\ & \lambda_2 y_1 = 0, \quad \lambda_3 y_2 = 0, \quad \lambda_4 y_3 = 0, \\ & x_1 + x_2 \leq 1, \\ & 2y_1 + y_2 - y_3 + x_1 - 2x_2 + 2 \leq 0, \\ & (x_1, x_2)^T \geq 0, \quad (y_1, y_2, y_3)^T \geq 0, \quad (\lambda_1, \lambda_2, \lambda_3, \lambda_4)^T \geq 0. \end{array}$$

Applying Algorithm 2.4, the solution of the above example is $(x_1, x_2, y_1, y_2, y_3)^T = (0.609, 0.391, 0, 0, 1.828)$, $f_u = 0.633887$, and $f_l = 1.6805$

Two-echelon supply chain system problem

Algorithm 2.4 is also applied to the problem was chosen from the engineering application called two-echelon supply chain system with one manufacturer and one retailer. The manufacturer purchases raw materials from the supplier first, then after the manufacturer's production and processing, the end products are sold to the retailer, this problem is formulated as bilevel models for joint pricing and lot-sizing decisions, see [15].

$$\begin{array}{ll} \max_{x_1,x_2} & f_u = (x_2 - P_s - T_c - M_c)x_1x_3y_1 - 0.5c_mTP_sx_3(y_1 - 1) - O_mx_1 \\ \textit{subject to} & P_s + T_c + M_c \leq x_2 \leq 10, \\ & x_1 \geq 0, \\ \max_{y_1,y_2} & f_l = x_1x_2x_3y_1(y_2 - 1) - 0.5c_rTx_2x_3 - O_rx_1y_1 \\ \textit{subject to} & 1 \leq y_2 \leq 5, \\ & y_1 \geq 0. \end{array}$$

where $T = 52$; $P_s = 4$; $T_c = 0.5$; $M_c = 1$; $c_m = c_r = 0.001$; $O_m = 400$; $O_r = 200$. For more details about the above application and its notations, see [15]. The above problem can be written as follows

$$\begin{array}{ll} \max_{x_1,x_2} & f_u = (x_2 - 5.5)x_1x_3y_1 - 0.104x_3(y_1 - 1) - 400x_1 \\ \textit{subject to} & 5.5 \leq x_2 \leq 10, \\ & x_1 \geq 0, \\ \max_{y_1,y_2} & f_l = x_1x_2x_3y_1(y_2 - 1) - 0.026x_2x_3 - 200x_1y_1 \\ \textit{subject to} & 1 \leq y_2 \leq 5, \\ & y_1 \geq 0. \end{array}$$

To apply our method, we write the above problem into the form (1.7) by introduce KKT conditions, which reads

$$\begin{array}{ll} \max & f_u = (x_2 - 5.5)x_1x_3y_1 - 0.104x_3(y_1 - 1) - 400x_1 \\ \textit{subject to} & x_1x_2x_3(y_2 - 1) - 200x_1 - \lambda_1 = 0, \\ & x_1x_2x_3y_1 + \lambda_2 - \lambda_3 = 0, \\ & \lambda_1 y_1 = 0, \\ & \lambda_2(y_2 - 5) = 0, \\ & \lambda_3(1 - y_2) = 0, \\ & 5.5 \leq x_2 \leq 10, \\ & 1 \leq y_2 \leq 5, \\ & x_1 \geq 0, \quad y_1 \geq 0, \quad (\lambda_1, \lambda_2, \lambda_3)^T \geq 0. \end{array}$$

We solve this model in case of the manufacturer is the leader, who makes the first decision, and the retailer is the follower. Our results, when applying Algorithm 2.4 is $x_1 = 5.878$, $x_2 = 6$, $x_3 = 19710.2412$, $y_1 = 7.7121$, $y_2 = 2.6009$, $f_u = 442353.28947$, and $f_l = 13931722.24717$. which is closed to whose reported in [15].

We present the numerical results of Algorithm (2.4) which have been performed on a laptop with Intel Core (TM)i7-2670QM CPU 2.2 GHz and 8 GB RAM. Algorithm (2.4) was implemented as a MATLAB code and run under MATLAB version 7.10.0.499 (R2013a).

Given a starting point $\bar{x}_0$. We chose $\delta_0 > 0$, $\delta_{min} = 10^{-3}$, and $\delta_{max} = 10^5\delta_0$. Also we chose $\eta_1 = 0.25$, $\eta_2 = 0.75$, $\alpha_1 = 0.5$, $\alpha_2 = 2$, $\epsilon_1 = 10^{-8}$, and $\epsilon_2 = 10^{-10}$.

5 Conclusions

The innovation for our algorithm for treating nonlinear bilevel programming problems is using a penalty method with trust-region strategy for solving such problem after converting it to a single level problem using the KKT conditions. This method maybe simpler than similar ideas and it does not need to compute a base of the null space. Also, the global convergence theorem for the introduced algorithm is presented. On comparing our algorithm with previous methods, the numerical results reflect the good behavior of the our algorithm and the computed results converge to the optimal solution. Finally, we apply our algorithm on one of the engineering application, called two-echelon supply chain system and the results was is closed to whose reported in literature.

References

[1] F. Al-Khayal, R. Horst, and P. Pardalos, Global optimization of concave functions subject to quadratic constraints: An application in nonlinear bilevel programming, Annals of Operations Research, 34(1992), 125-147.

[2] J. Dennis, M. El-Alem, and K. Williamson, A trust-region approach to nonlinear systems of equalities and inequalities, SIAM J Optimization, 9(1999) 291 - 315.

[3] M. El-Alem, A global convergence theory for a class of trust-region algorithms for constrained optimization, PhD thesis, Department of Mathematical Sciences, Rice University, Houston, Texas, 1988.

[4] T. Edmunds and J. Bard, Algorithms for nonlinear bilevel mathematical programs, IEEE transactions on Systems, Man, and Cybernetics, 21(1991), 83-89.

[5] B. El-Sobky, A Multiplier active-set trust-region algorithm for solving constrained optimization problem, Applied Mathematics and Computation, 219(2012) 127 - 157.

[6] B. El-Sobky, A New Convergence theory for Trust-Region Algorithm for Solving Constrained Optimization Problems, Applied Mathematics Science, 7(110)(2013) 5469 - 5489.

[7] B. El-Sobky, An interior-point penalty active-set trust-region algorithm, Journal of the Egyptian Mathematical Society, 24(2016) 672-680.

[8] B. El-Sobky and Y. Abouel-Naga, Multi-objective optimal load flow problem with interior-point trust-region strategy, Electric Power Systems Research 148(2017) 127 - 135.

[9] A. Fiacco and G. McCormick, Nonlinear programming: Sequential unconstrained minimization techniques, John Wiley and Sons, New York, (1968).

[10] R. Fletcher, An l_1 penalty method for nonlinear constraints, In Numerical Optimization, SIAM, (1984), 26-40.

[11] J. Falk and J. Liu, On bilevel programming, Part I: general nonlinear cases, Mathematical Programming, 70(1)(1995), 47-72.

[12] H. Gumus and A. Flouda, Global Optimization of Nonlinear Bilevel Programming Problems, Journal of Global Optimization, 20(2001) 1-31.

[13] O. Mangasarian. Nonlinear Programming. McGraw-Hill Book Company, New York (1969).

[14] D. Muu and N. Quy A Global Optimization Method for Solving Convex Quadratic Bilevel Programming Problems, Journal of Global Optimization, 26(2003) 199219.

[15] W. Ma, M. Wang, and X. Zhu, Improved particle swarm optimization based approach for bilevel programming problem-an application on supply chain model, Int. J. Mach. Learn.and Cyber 5(2014), 281-290.

[16] L. Niu and Y. Yuan, A new trust region algorithm for nonlinear constrained optimization, Journal of Computational Mathematics 28(2010), 72-86.

[17] Y. Pei and D. Zhu, A trust-region algorithm combining line search filter technique for nonlinear constrained optimization, International Journal of Computer Mathematics, 91(8)(2014) 1817 - 1839.

[18] T. Steihaug, The conjugate gradient method and trust region in large scale optimization. SIAM J.Numer. Anal., 20(1983) 626-637.

[19] G. Savard and J. Gauvin, The steepest descent direction for the nonlinear bilevel programming problem, Operations Research Letters, 15(1994) 265-272.

[20] N. Thoai, Y. Yamamoto, and A. Yoshise, Global optimization method for solving mathematical programs with linear complementarity constraints, Discussion Paper No. 987, Institute of Policy and Planning Sciences, University of Tsukuba, Japan, May 2002.

[21] X. Wang and Y. Yuan, A trust region method based on a new affine scaling technique for simple bounded optimization, Optimization Methods and Software 28(4)(2013), 871-888.

[22] X. Wang and Y. Yuan, An augmented Lagrangian trust region method for equality constrained optimization, Optimization Methods and Software, 30(3)(2015), 559-582.

[23] Y. Yuan, On the convergence of a new trust region algorithm. Numer. Math., 70(1995), 515-539 .

[24] Y. Yuan, Recent advances in trust region algorithms, Math. Program. Ser. B, 151(2015) 249 - 281.