Algorithms for Asymptotically Exact Minimizations in Karush-Kuhn-Tucker Methods

Koudi Jean¹, Guy Degla ¹, Babacar Mbaye Ndiaye² & Mamadou Kaba Traoré³

¹ Institute of Mathematics and Physical Sciences, University of Abomey Calavi, Porto-Novo, Benin
² Laboratory of Mathematics of Decision and Numerical Analysis, University of Cheikh Anta Diop, Dakar, Senegal
³ Computer Laboratory, ISIMA (University of Blaise Pascal), Clermont-Ferrand, France

Correspondence: Babacar Mbaye Ndiaye, Laboratory of Mathematics of Decision and Numerical Analysis. University of Cheikh Anta Diop. BP 45087 Dakar-Fann, 10700, Dakar, Senegal. E-mail: babacarm.ndiaye@ucad.edu.sn

Received: January 2, 2018 Accepted: January 22, 2018 Online Published: February 19, 2018
doi:10.5539/jmr.v10n2p36 URL: https://doi.org/10.5539/jmr.v10n2p36

Abstract

We provide two new algorithms with applications to asymptotically exact minimizations with inequalities constraints. These results generalize and improve the works of Andreani, Birgin, Martinez and Schuverdt on minimization with equality constraints. Numerical examples show that our proposed analysis gives convergence results.

Keywords: nonlinear programming, augmented lagrangian methods, numerical experiments, approximate KKT point

1. Introduction

The Kuhn-Tucker condition is often used to obtain important results in economics, especially in decision problems that occur in static situations, for example to show the existence of a balance for a competitive economy, main agents constraints and so on. Kuhn-Tucker conditions for the optimization problem under inequality and equality constraints have a global shape that naturally incorporate the Lagrange multiplier method (introduced by Lagrange in 1788). The application of this method to an optimization problem under constraint leads to the resolution of the Karush, Kuhn and Tucker (KKT) system.

Throughout this work, we consider on \( \mathbb{R}^n \) \((n \in \mathbb{N})\) the ordering relation \( \leq \) defined by:

\[ \forall u \in \mathbb{R}^n, \forall l \in \mathbb{R}^n, u \leq l \iff [u]_i \leq [l]_i, \forall i \in \{1; \ldots; n\} \] where \([u]_i\) is the \(i\)th component of the row \(u\). We also consider the operator projection \(P_+ : \mathbb{R}^n \rightarrow \mathbb{R}^+_n\) by

\[ [P_+ (u)]_i = \begin{cases} [u]_i & \text{if } [u]_i \geq 0 \\ 0 & \text{otherwise} \end{cases} \]

We shall study in this work the optimization problem of type

\[
(P) : \min_{x \in K} f(x)
\]

with \(K = \{ x \in \mathbb{R}^n; g_i(x) \leq 0 \forall i \in I = \{1, \ldots, m\} \}

Under Karush-Kuhn-Tucker constraint qualification, when the problem (1) is differentiable (i.e., all the involved functions are differentiable), a first-order condition for a point \(x^*\) to be optimal is that it satisfies the following system:

\[
\begin{align*}
\nabla f(x) + \sum_{i=1}^{n} \lambda_i \nabla g_i(x) &= 0 \\
\lambda_i g_i(x) &= 0 \quad \forall i \in I \text{ (exclusion condition)} \\
\lambda_i &\geq 0 \quad \forall i \in I
\end{align*}
\]

When the constraints of the problem are equality constraints, the exclusion condition in (3) becomes obvious. So, to find a solution of the problem (1) is to find a point satisfying the system (3) without the exclusion condition. The manual resolution of this system becomes complicated especially when the size of the problem becomes large.

In April 1991, Andrew R. C. et al. published two algorithms in (Andrew, R.C. & et al., 1991) for solving KKT systems arising from differentiable optimization problems under equality constraints \(K\) defined by:

\[
K = \{ x \in \mathbb{R}^n; h_i(x) = 0 \forall i \in I; u \leq x \leq l \}, \ u \in \mathbb{R}^n \text{ and } l \in \mathbb{R}^n
\]
These algorithms are based on the augmented Lagrangian defined by:

\[ L(x, \lambda, S; \mu) = f(x) + \sum_{i \in I} \lambda_i h_i(x) + \sum_{i \in I} S_i \frac{\mu}{2} (h_i(x))^2 \]  

where \( \mu \in \mathbb{R}^n_+ \) and \( S \) is an invertible diagonal matrix such that \( 0 < S_{ii} \).

In 2006, Andreani R. et al., in (Andreani, R. & et al., 2006) have taken up these algorithms by posing \( \rho = \frac{\mu}{\nu} \) called penalty parameter. Their algorithms are also based on the augmented lagrangian defined by:

\[ L(x, \lambda, \rho) = f(x) + \sum_{i \in I} \lambda_i g_i(x) + \frac{1}{2} \sum_{i \in I} \rho_i (g_i(x))^2 \]  

If all the functions of the constraints are differentiable and if the objective function is differentiable, we have:

\[ \nabla_s L(x, \lambda, \rho) = \nabla f(x) + \sum_{i \in I} (\lambda_i + \rho_i g_i(x)) \nabla_i g_i(x) \]  

The principle of these algorithms is to find \( x \in K \) such that

\[ \| P_\Omega [x - \nabla_s L(x, \lambda, \rho)] - x \|_\infty = 0 \]  

that is to say

\[ -\nabla_s L(x, \lambda, \rho) \in T^0_\Omega(x) \]  

Where \( P_\Omega \) is the projection operator on \( \Omega = \{ x \in \mathbb{R}^n : lb \leq x \leq ub \} \)

In our work, we improve these algorithms in order to adapt them to optimization problems under inequality constraints. Our algorithm guarantees constraint qualifications at the end point of sequence generated by each of these algorithms (and satisfaction of exclusion condition). We modify the estimation of Lagrange multipliers and add a new condition for the resolution of a sub-problem in order to determine the approximate solutions \( x_k \) at each iteration \( k \). We present the foundations of this algorithm including a convergence analysis result.

The rest of the paper is organized as follows. In section 2, we review part of literature on KKT algorithm, followed by an analysis of our algorithm and the obtained results in section 3. Finally, we conclude with prospective recommendations in section 4.

2. Some Preliminaries

2.1 Admissible Direction and Tangent Cone

Let \( K \) be a feasible set of the problem (P) and \( x_0 \) be an admissible element.

- An admissible direction at \( x_0 \) is any vector tangent to an arc of curve (sufficiently regular) admissible in \( x_0 \). In other words, it is any element \( d \) such that there exists \( (x_0) \in K^n \rightarrow x_0, \epsilon_n \rightarrow 0 \) and \( \frac{x_0 - x_0}{\epsilon_n} \rightarrow d \).

Let \( \omega_n = \frac{x_0 - x_0}{\epsilon_n} \), we obtain \( \omega_n \rightarrow d \) and \( \epsilon_n \omega_n + x_0 \in K \forall n \)

- The set of all admissible directions is called the tangent cone at \( x_0 \) of \( K \) and denote by \( T_K(x_0) \)

- The polar cone at \( x_0 \) of \( K \) is

\[ T^0_K(x_0) = \left\{ u : \langle u, d \rangle \leq 0 \ \forall \ d \in T_K(x_0) \right\} \]

Suppose that

\[ K = \left\{ x \in X : g(x) \leq 0, \ \forall i \in I \right\} \]

- The linear tangent cone at \( x_0 \) of \( K \) denote by \( T^l_K(x_0) \) is defined by

\[ T^l_K(x_0) = \left\{ d : \nabla g_i(x) \cdot d \leq 0, \ \forall i \in I(x_0) \right\} \]

and its linear polar cone at \( x_0 \) is

\[ (T^l_K)^0(x_0) = \left\{ \sum_{i \in I(x_0)} \lambda_i \nabla g_i(x_0) : \lambda_i \geq 0, \ \forall i \in I(x_0) \right\} \]
where \( I(x_0) = \{ i \in I; g_i(x_0) = 0 \} \) It is easy to see that:

\[
T_K(x_0) \subseteq T^\text{lin}_K(x_0) \quad (10)
\]

and

\[
(T^\text{lin}_K)^0(x_0) \subseteq T^0_K(x_0) \quad (11)
\]

Recall that the constraint \( K \) is qualified at \( x_0 \) if

\[
T_K(x_0) = T^\text{lin}_K(x_0) \quad (12)
\]

Among the multitudes sufficient conditions for the qualification of the constraint, we can mention those such as: 

\textit{Slater (1950) and Karlin (1959) constraints}: that often apply in nondifferentiable cases for convex problems.

\textit{AHUCQ Arrow, Hurwicz and Uzawa Constrained qualification}.

\textit{MFCQ Mangasarian and Fromovitz Constrained qualification}: it guarantees the existence of Lagrange multiplier satisfying the system of Karush Kuhn and Tucker at the optimum.

\textit{LICQ Linearly Independent Constraint Qualification Gerd Wachsmuth, 2013}: which is the strongest of all the qualification constraints that apply to differentiable problems; it guarantees the existence and uniqueness of Lagrange multiplier satisfying the system of Karush Kuhn and Tucker at the optimum.

\textit{(CPLD) Constant Positive Linear Dependence condition Qi, L. & Wei, Z., 2000; Definition 2.1}:

Let \( A = \{ a^1, ..., a^m \} \) and \( B = \{ b^1, ..., b^q \} \) be families of elements of \( \mathbb{R}^n \) such that \( A \cup B \) is an empty set. \( A \) and \( B \) are said to be positively linearly dependent if there exists \( a \in \mathbb{R}^m \) and \( \beta \in \mathbb{R}^q \) such that \( (a; \beta) \neq (0; 0) \) and

\[
\sum_{i=1}^{m} \alpha_i a^i + \sum_{j=1}^{q} \beta_j b^j = 0 \quad (13)
\]

Otherwise, we say that \( A \) and \( B \) are positively linearly independent. We say that \( x^* \) satisfies the qualification constraint \((CPLD)\), if there exists \( I_1 \subset I(x^*) = \{ i \in I; g_i(x^*) = 0 \} \), \( J_1 \subset J \) such that the family \( \{ \nabla g_i(x^*) \}_{i \in I_1} \cup \{ \nabla h_j(x^*) \}_{j \in J_1} \) is positively linearly dependent, if there exists a neighborhood \( V(x^*) \) such that \( \forall x \in V(x^*) \) the family \( \{ \nabla g_i(x) \}_{i \in I_1} \cup \{ \nabla h_j(x) \}_{j \in J_1} \) is linearely dependent.

2.2 \textit{KKT Point}

We say that a feasible point is a Karush-Kuhn-Tucker point of the problem \((1)\) if it checks the system \((3)\) called Karush-Kuhn-Tucker (KKT) system.

A KKT point is not necessarily a minimum point, it is usually a stationary point (minimum, or maximum, or saddle point). The determination of such a point consists in solving the system \((3)\). Our goal is to determine from these algorithms, the points of KKT which allow to have the exact optimum value. But it is sometimes difficult to reach precisely such a point.

2.3 \textit{Approximate KKT Point}

Consider the optimization problem defined by

\[
(P) : \min_{x \in K} f(x) \quad \text{where} \quad K = \{ x \in \mathbb{R}^n ; \ g_i(x) \leq 0 \ \forall \ i \in I = \{ 1, ..., m \}, h_j(x) = 0, \ \forall \ j \in \{ 1, \cdots, q \} \} \quad (14)
\]

and all the functions are differentiable.

A feasible point \( x^* \) is said to be approximate KKT point of \((P)\) if there exists a sequence \((x^k)_k \subset \mathbb{R}^n\) that converges to \( x^* \), a sequence \((\lambda^k)_k \subset \mathbb{R}^n \), a sequence \((\mu^k)_k \subset \mathbb{R}^q \) and a sequence \((\epsilon^k)_k \subset \mathbb{R}_+\) converging to zero such that

\[
\left\{ \begin{array}{l}
\nabla f(x^k) + \sum_{i=1}^{m} \lambda^k_i \nabla g_i(x^k) + \sum_{j=1}^{q} \mu^k_j \nabla h_j(x^k) \\
\lambda^k_i (g_i(x^k) + \epsilon^k) = 0 \ \forall \ i \in I \ \text{et} \ \forall \ k
\end{array} \right. \to 0 \quad \text{when} \ k \to \infty
\]

\[
\text{Proposition 2.1.} \ (\text{See Qi, L. & Wei, Z., 2000; Theorem 2.7})
\]

\textit{Let} \( x^* \) \textit{be an approximate KKT point that satisfies the (CPLD) condition, then} \( x^* \) \textit{is a KKT point.}
2.4 Some Reminders on Optimality Conditions

An immediate consequence of the local optimality of \( x^* \) for (1) is

\[
\nabla f(x^*) \cdot d \geq 0 \quad \forall \ d \in T_K(x^*)
\]

(16)

that is to say

\[
-\nabla f(x^*) \in (T_K^o(x^*))
\]

(17)

with \( K = \{ x \in X : g_i(x) \leq 0, \forall i \in I \} \). According to KKT condition, the relation (17) implies

\[
\sum_{i \in I} \lambda_i \nabla g_i(x^*) \in T_K^o(x^*)
\]

(18)

3. Main Results

3.1 Algorithms

Let us first recall that any constrained optimization problem whose feasible set is defined by

\[
K = \{ x \in \mathbb{R}^n, g_i(x) \leq 0 \ \forall \ i \in I = 1, 2, ..., m, \ h_j(x) = 0 \ \forall \ j \in J = 1, 2, ..., q \}
\]

(19)

can be transformed into a problem of the same type as (1) where all the constraints are inequality constraints. This is because each equality constraint can be transformed into double inequalities. That is to say

\[
h(x) = 0 \iff h(x) \leq 0 \quad \text{and} \quad h(x) \geq 0
\]

(20)

Our algorithms are based on the augmented lagrangian defined by:

\[
L(x, \lambda, \mu) = f(x) + \sum_{i \in I} \lambda_i g_i(x) + \frac{1}{2} \sum_{i \in I} \rho_i (g_i(x))^2
\]

(21)

Like R. Andreani et al. in (Andreani, R. & et al., 2006), We shall compute at each iteration \( k \) a point \( x_k \) such that

\[
\|P_{\Omega} [x_k - \nabla_x L(x_k, \lambda_k, \rho_k)] - x_k\|_\infty \leq \varepsilon_k
\]

(22)

where \( \varepsilon_k \) is a decreasing sequence converging to zero. The point \( x_k \) at iteration \( k \) for (22) constitutes a non-obvious subproblem to solve in the algorithm. The following proposition allows us to easily construct a sequence of points under certain hypotheses satisfying (22)

Proposition 3.1. Let \( \Omega \) be a close subset of \( \mathbb{R}^n \), \( \lambda \in \mathbb{R}^m \), \( \rho \in \mathbb{R}^q \), and \( (x_k)_k \) be a sequence defined by:

\[
x_{k+1} = P_{\Omega} [x_k - h_k \nabla_x L(x_k, \lambda_k, \rho_k)]
\]

(23)

where \( h_k \) is a sequence that converges to zero and \( \sum_{k \geq 0} h_k < +\infty \). Then the sequence \( \|P_{\Omega} [x_k - \nabla_x L(x_k, \lambda_k, \rho_k)] - x_k\|_\infty \) converge.

Proof

To simplify, let us denote \( \nabla_x L(x_k, \lambda_k, \rho_k) \) by \( \nabla_x L(x_k) \). By definition \( x_k \in \Omega \) for all \( k \). The projection operator is 1-lipschitzian, thus

\[
\|x_k - x_{k+1}\| = \|x_k - P_{\Omega} [x_k - h_k \nabla_x L(x_k)]\| = \|P_{\Omega} [x_k] - P_{\Omega} [x_k - h_k \nabla_x L(x_k)]\| \leq \|x_k - (x_k - h_k \nabla_x L(x_k))\| = \|h_k \nabla_x L(x_k)\|
\]

(24)

(25)

(26)

Let \( U_k = P_{\Omega} [x_k - \nabla_x L(x_k, \lambda_k, \rho_k)] - x_k \). Again the 1-Lipschitzness of the projection we have:

\[
\|U_{k+1} - U_k\| = \|x_{k+1} - x_k - (P_{\Omega} [x_{k+1} - \nabla_x L(x_{k+1})] - P_{\Omega} [x_k - \nabla_x L(x_k)])\| \\
\leq \|x_{k+1} - x_{k+1}\| + \|x_k - x_{k+1} - (\nabla_x L(x_{k+1}) - \nabla_x L(x_k))\| \leq 2 \|x_k - x_{k+1}\| + \|\nabla_x L(x_{k+1}) - \nabla_x L(x_k)\|
\]

(27)

(28)

(29)
(x_k)_{k} being a sequence of descent towards the optimum, there exists M ≥ 0 such that \(\|\nabla_x L(x_k)\| \leq M\) and \(\|\nabla_x L(x_{k+1}) - \nabla_x L(x_k)\| \leq M\|x_{k+1} - x_k\|\) then

\[
\|U_{k+1} - U_k\| \leq (2 + M)h_k
\]  

(30)

Let \(p, q \in \mathbb{N}\) (suppose that \(p \geq q\)) we have:

\[
\|U_p - U_q\| \leq \sum_{s=0}^{p-q-1} \|U_{p-s} - U_{p-s-1}\|
\]  

(31)

According to (30) we have:

\[
\|U_p - U_q\| \leq \sum_{s=q}^{p+1} (2 + M)h_s = (2 + M) \sum_{s=q}^{p+1} h_s
\]  

(32)

As \(h_k \rightarrow 0\) and \(\sum_{k=0}^{\infty} h_k < +\infty\), then \(U_k\) are Cauchy sequence. Then it converge because \(\mathbb{R}^n\) is a complete space.

The convergence towards zero will be studied in these propositions to follows.

As constraints are inequalities, we shall add a new condition defined by:

\[
\sum_{i \in I} | [\lambda_i] g_i(x_k) | \leq \epsilon_k \quad \forall \ k
\]  

(33)

in order to satisfy at the end of each algorithm the exclusion condition in the KKT system. This condition will serve to converge rapidly towards the optimum. In fact, it is easy to see that

\[
\sum_{i \in I} | [\lambda_i] g_i(x_k) | = 0 \iff [\lambda_i] g_i(x_k) = 0 \quad \forall \ i \in I
\]  

(34)

In 2006, Andreani R. et al., in (Andreani R. & et al., 2006) have defined the projection on \(\Omega = \{x \in \mathbb{R}^n, \ lb \leq x \leq ub\}\). The projection on this set can give inadmissible points specially when \(lb\) and \(ub\) are not well defined according to the admissible set (eg when they are away from the permissible assembly). To approximate advantage the projected to the admissible set we shall define the projection on the set

\[
\Omega(x_k) = \{x \in \mathbb{R}^n, Ax \leq b, lb \leq x \leq ub\} \quad \text{were} \quad A = \nabla g(x_k) \quad \text{and} \quad b_i = (\nabla g_i(x_k))^T x_k - g_i(x_k) \quad \forall \ i
\]  

(35)

Indeed a point \(x \in \mathbb{R}^n\) will be admissible if and only if \(g_i(x) \leq 0 \quad \forall \ i \in \{1, ..., m\}\). Since all the functions are differentiable, at the iteration \(k + 1\) knowing that \(x_k\) is already determined, \(g_i(x_{k+1}) - g_i(x_k) = \nabla g_i(x_k)(x_{k+1} - x_k) + \|x_{k+1} - x_k\| g_i(x_{k+1} - x_k)\) were

\[
\lim_{\|x_{k+1} - x_k\| \rightarrow 0} \frac{g_i(x_{k+1} - x_k)}{\|x_{k+1} - x_k\|} = 0.
\]

Thus, approximately we have \(g(x_{k+1}) \leq 0 \iff \nabla g_i(x_{k+1})^T x_k - g_i(x_k) \leq \nabla g_i(x_k)^T x_k - g_i(x_k)\).

At each iteration it is necessary to solve a quadratic problem defined by

\[
(Q) : \begin{cases} 
\min \|u - x\|^2 \\
(\nabla g_i(x_k))^T x \leq (\nabla g_i(x_k))^T x_k - g_i(x_k), \quad 1 \leq i \leq m \\
\text{lb} \leq x \leq \text{ub} \\
x \in \mathbb{R}^n
\end{cases}
\]  

(36)

It is clear that \(K \subset \Omega(x_k) \subset \Omega \quad \forall \ k\). Then for all \(y \in \Omega\) and \(x \in \Omega(x_k)\), we have \(d(y, K) \geq d(x, K)\).

Also in the stopping criterion we shall impose that \(g_i(x) \leq 0 \quad \forall \ i \in I\)

We propose the two following algorithms \(C_1\) and \(C_2\).

**Algorithm** \(C_1\)

\(x_0 \in K, \ \tau \in [0; 1], \ \gamma > 1, \ \rho_0 \in \mathbb{R}^m_{++} \) such that \([\rho_0]_i = \|\rho_0\|_\infty \quad \forall i \in I, \ \ (\varepsilon_2)_k \in \mathbb{R}_+ \) a decreasing sequance converging to 0 and \(0 < \varepsilon^* \ll 1\)
Step 1: Evaluate $\lambda_0$

$$\lambda_0 \in \text{Argmax}\{L(x_0, \lambda), \lambda \geq 0\}$$  \hspace{1cm} (37)

Step 2: (Estimate multipliers) If

$$\|P_{\Omega,x_k}[x_k - \nabla_x L(x_k, \lambda_k, \rho_k)] - x_k\| \leq \varepsilon^* \text{ and } g_i(x_k) \leq \varepsilon^* \forall i \in I$$  \hspace{1cm} (38)

stop
Otherwise:

$$[\lambda_{k+1}]_i = P_+([\lambda_k]_i + [\rho_k]_i [g(x_k)]_i) \forall i \in I$$  \hspace{1cm} (39)

Step 3: (Update the penalty parameters)

We define the set

$$\Gamma_k = \{i \in I, \ |g_i(x_k)\| \geq \tau \|g(x_{k-1})\|_\infty\}$$  \hspace{1cm} (46)

If $\Gamma_k = \emptyset$ then $\rho_{k+1} = \rho_k$
Otherwise

$$[\rho_{k+1}]_i = \gamma [\rho_k]_i \text{ if } i \in \Gamma_k \text{ and } [\rho_{k+1}]_i = [\rho_k]_i \text{ if } i \notin \Gamma_k$$
Step 4: Solving the subproblem:

\[ \varepsilon_{k+1} = \min \left\{ \varepsilon_{k+1} : \sum_{i \in I} \frac{|g_i(x_k)|}{1 + |\rho_k_i||g_i(x_k)|} \right\} \] (47)

We check \( x_{k+1} \) such that

\[ \|P_{\Omega(x_k)}[x_{k+1} - \nabla L(x_{k+1}, \lambda_{k+1}, \rho_{k+1})] - x_{k+1}\|_\infty + \sum_{i \in I} \|\lambda_{k+1}, g_i(x_{k+1})\| \leq \varepsilon_{k+1} \] (48)

Step 5: \( k \leftarrow k + 1 \)

Return to Step 2

Recall that the resolution of the sub-problem in each algorithm is based on the proposition (3.1). The point \( x_0 \) can also be unfeasible point; in this case we project it in the feasible set \( \Omega \) and obtain an approximate feasible point.

3.2 Theorical Results

Note that, at the end of each algorithm, the sequence \((x_k)_k\) converge or have a subsequence that converges to a point \( x^* \) according to (3.1).

According to (Theorem 3 (Andreani R. & et al., 2006)), Andreani et al. (2006) have proved that if \( x^* \) is feasible and satisfies the CPLD constraint qualification then it is an approximate KKT point. Through the following propositions we will give other conditions under which \( x^* \) will be a KKT point

**Proposition 3.2.** Assume that the sequence \((x_k)_k\) is generated by algorithm C1 and that \( x^* \) is a limit point. Then

\[ \sum_{i \in I} (-g_i(x^*)) \cdot \nabla g_i(x^*) \in T^0_{\Omega}(x^*) \] (49)

and if \( x^* \) is feasible, then there exists \( \lambda_i \geq 0 \; \forall \; i, \) such that

\[ \sum_{i \in I(x^*)} \lambda_i \cdot \nabla g_i(x^*) \in T^0_{\Omega(x^*)}(x^*) \] (50)

**Proof**

Remark that

\[ \|P_{\Omega(u + tv)} - u\|_\infty \leq \|P_{\Omega(u + v)} - u\|_\infty \; \forall \; t \in [0; 1] \; \text{and} \; \forall \; u, v \in \mathbb{R}^n \] (51)

Let \( \overline{\rho_k} = \|\rho_k\|_\infty \; \forall \; k \)

By Step 3, if the sequence \((\rho_k)_k\) is bounded then there exists \( k_0 \) such that for all \( k \geq k_0, \; \|g(x_k)\| \leq \varepsilon_k \) which makes it possible to conclude that the limit \( x^* \) is feasible.

Otherwise, \( \overline{\rho_k} \rightarrow \infty \) and:

\[ P_{\Omega(x_k)}(x_k - \nabla L(x_k, \lambda_k, \rho_k)) - x_k = \left[ P_{\Omega(x_k)} \left( x_k - \left( \nabla f(x_k) + \sum_{i \in I} (\lambda_i, + \overline{\rho_k} g_i(x_k)) \nabla g_i(x_k) \right) \right) \right] - x_k \] (52)

Let \( u = x_k, \; t = \frac{1}{\overline{\rho_k}}, \) and \( v = -\left( \nabla f(x_k) + \sum_{i \in I} (\lambda_i, + \overline{\rho_k} g_i(x_k)) \nabla g_i(x_k) \right) \) we have

\[ \left\| P_{\Omega(u,v)} \left( x_k - \left( \nabla f(x_k) + \sum_{i \in I} (\lambda_i, + \overline{\rho_k} g_i(x_k)) \nabla g_i(x_k) \right) \right) \right\|_\infty \leq \left\| P_{\Omega(x_k)} (x_k - \nabla L(x_k, \lambda_k, \rho_k)) - x_k \right\|_\infty \] (53)

Hence, for \( k \) sufficiently large, we have \( \overline{\rho_k} \rightarrow \infty \) and \( \left| P_{\Omega(x_k)} (x_k - \nabla L(x_k, \lambda_k, \rho_k)) - x_k \right|_\infty \rightarrow 0 \) (according to (48)).

Then

\[ \left\| P_{\Omega(x^*)} \left( x^* - \sum_{i \in I} g_i(x^*) \cdot \nabla g_i(x^*) \right) - x^* \right\|_\infty = 0 \] (54)
In this case we have the following proposition:

**Proposition 3.3.** Assume that the sequence \((x_k)_k\) is generated by algorithm C_2 and that \(x^*\) is a limit point. If there exists \(k_0\) such that \(\Gamma_k\) is an empty set \(\forall k \geq k_0\), then \(x^*\) is a feasible point.

**Proof**

By Step_3 in algorithm C_2, we have:

\[
\|g(x_{k+1})\| < \tau \|g(x_k)\| \quad \text{(58)}
\]
\[
\|g(x_{k+2})\| < \tau \|g(x_{k+1})\| \quad \text{(59)}
\]
\[
\|g(x_{k+1})\| < \tau \|g(x_k)\| \quad \text{(60)}
\]
\[
\|g(x_{k+m})\| < \tau^m \|g(x_k)\| \quad \text{(64)}
\]

When \(m \to +\infty\), we have \(\|g(x^*)\| = 0\), because \(0 \leq \tau < 1\).

Then \(g(x^*) = 0\), that is to say \(x^*\) is feasible and all constraints are active at \(x^*\). Note that in the case where the number of constraints is high, it is almost impossible to have all the constraints to be active at a point.

**Proposition 3.4.** Assume that the sequence \((x_k)_k\) is generated by algorithm C_2 and that \(x^*\) is a limit point. Then either \(x^*\) is feasible, or there exists \(\omega \in \mathbb{R}^n_+\) such that

\[
\sum_{i \in I} \omega_i (-g_i(x^*)) \cdot \nabla g_i(x^*) \in T^0_{\Omega(x^*)}(x^*)
\]

And if \(x^*\) is feasible, then there exists \(A_i \geq 0 \ \forall i\) such that

\[
\sum_{i \in I} A_i \cdot \nabla g_i(x^*) \in T^0_{\Omega(x^*)}(x^*)
\]

**Proof**

By the proposition (3.3), if there exists \(k_0\) such that \(\Gamma_k = \emptyset\), \(\forall k \geq k_0\), then \(x^*\) feasible and \(g(x^*) = 0\).

Otherwise, \(\|\rho_k\| \to \infty\).

By Step_4 we have

\[
\|P_\Omega[x_{k+1} - \nabla_x L(x_{k+1}, \lambda_{k+1}, \rho_{k+1})] - x_{k+1}\| \leq \varepsilon_{k+1}
\]
which means that
\[ P_{\Omega(x_k)}(x_k - (\nabla f(x_k) + \sum_{i \in I} [\lambda_k]_i + [\mu_k]_i \nabla g_i(x_k))) \leq e_k \] (68)

Hence by relation (51) we obtain
\[ \left\| P_{\Omega(x_k)} \left( x_k - \left( \nabla f(x_k) + \sum_{i \in I} \frac{[\lambda_k]_i}{\| \lambda_k \|} + \frac{[\mu_k]_i}{\| \mu_k \|} \nabla g_i(x_k) \right) \right) \right\|_{\infty} \leq \| P_{\Omega(x_k)} [x_k - \nabla_s L(x_k, \lambda_k, \mu_k)] - x_k \|_{\infty} \leq e_k \] (69)

As \( k \to +\infty \), we have \( \left( \frac{\nabla f(x_k)}{\| \lambda_k \|} \right)_k \to 0 \), \( \left( \frac{\lambda_k}{\| \lambda_k \|} \right)_k \to \lambda^* \) and so:
\[ \left\| P_{\Omega(x_k)} \left[ x^* - \sum_{i \in I} \omega^*_i g_i(x^*) \nabla g_i(x^*) \right] - x^* \right\| = 0 \] (70)

Thus
\[ \left\{ - \sum_{i \in I} \omega^*_i g_i(x^*) \nabla g_i(x^*) \right\} = \left\{ x^* - \sum_{i \in I} \omega^*_i g_i(x^*) \nabla g_i(x^*) \right\} = \Omega^0_{\Omega(x^*)}, \text{ that is to say} \]
\[ \left\{ \sum_{i \in I} \omega^*_i (-g_i(x^*)) \cdot \nabla g_i(x^*) \right\} \in T^0_{\Omega(x^*)}(x^*) \] (71)

Hence if \( x^* \) is feasible \( -g_i(x^*) \geq 0 \ \forall \ i \), then there exists \( \lambda_i \geq 0 \ \forall \ i \) such that
\[ \sum_{i \in \Omega(x^*)} \lambda_i \cdot \nabla g_i(x^*) \in T^0_{\Omega(x^*)}(x^*) \] (72)

Note that the results of propositions (3.2) and (3.4) are an implication of the optimality condition defined in. Thus, they do not allow us to assert the optimality of the limit given by each algorithm. The following theorem 3.1 will give us a sufficient condition for the limit to be an optimal point.

We note that the convergence towards an admissible point of the algorithms depends on the convergence of the penalty parameters.

**Theorem 3.1. (Convergence to an optimal point)**

Assume that the sequence \( (x_k)_k \) is generated by algorithm \( C_1 \) or \( C_2 \) and that a limit point \( x^* \) is a feasible point. Then there exists \( \overline{\lambda} \in \mathbb{R}_+^m \) and \( \rho^* \in \mathbb{R}_+^n \) such that \( \nabla_s L(x^*, \overline{\lambda}, \rho^*) = 0 \). That is to say \( x^* \) is an optimal point and the associated Lagrange multiplier is \( \overline{\lambda} \).

**Proof**

Let us recall that at each iteration \( k \), the set \( \Omega(x_k) \) being a closed convex set, the projection probem
\[ \min_{v \in \Omega(x_k)} \frac{1}{2} \| v - (x_k - \nabla_s L(x_k, \lambda_k, \rho_k)) \|_2^2 \] (73)

has unique solution \( v_k = P_{\Omega(x_k)}[x_k - \nabla_s L(x_k, \lambda_k, \rho_k)] \). Apply the KKT conditions to (73), there exists \( \overline{\lambda}_k \in \mathbb{R}_+^m \), \( \mu_k^i \in \mathbb{R}_+^n \) and \( \mu_k^e \in \mathbb{R}_+^e \) such that:
\[ \begin{cases} v_k - x_k + \nabla_s L(x_k, \lambda_k, \rho_k) + \sum_{i=1}^m [\overline{\lambda}_k]_i \nabla g_i(x_k) + \sum_{j=1}^n [\mu_k^e]_j e_j - \sum_{j=1}^n [\mu_k^e]_j e_j = 0 \\ [\overline{\lambda}_k]_i \cdot [\nabla g_i(x_k) - \nabla g_i(x_k)] = 0 \ \forall \ i \\ [\mu_k^i]_i (x_k - x_k) + [\mu_k^e]_j (e_j - [\mu_k^e]_j) = 0 \ \forall \ j \end{cases} \] (74)

By Step 4 \( \| v_k - x_k \| = \| P_{\Omega(x_k)}[x_k - \nabla_s L(x_k, \lambda_k, \rho_k)] - x_k \| \leq e_k \) which implies
\[ \lim_{k \to \infty} \| v_k - x_k \| = 0 \] (75)

Hence by (74)
\[ x_k - v_k = \nabla_s L(x_k, \lambda_k, \rho_k) + \sum_{i=1}^m [\overline{\lambda}_k]_i \nabla g_i(x_k) + \sum_{j=1}^n [\mu_k^i]_j e_j - \sum_{j=1}^n [\mu_k^e]_j e_j \] (76)
\[ = \nabla_s L(x_k, \overline{\lambda}_k, \rho_k) + \sum_{j=1}^n [\mu_k^e]_j e_j - \sum_{j=1}^n [\mu_k^e]_j e_j \] (77)
were $\bar{\lambda}_k = \lambda_k + \bar{\lambda}_k$

Hence, according to (75) we have

$$\lim_{k \to \infty} \left[ \nabla_x L(x_k, \bar{\lambda}_k, \rho_k) + \sum [\mu^*_i]_i e_i - \sum [\mu^*_j]_j e_j \right] = 0 \quad (78)$$

Let us denote $\Omega = \{ x \in \mathbb{R}^n \mid lb \leq x \leq ub \}$ it is necessary to choose $lb$ and $ub$ such that $K \subset int(\Omega)$ where $int(\Omega)$ denote the interior of $\Omega$. Since $x_k \to x^*$ and $\| v_k - x_k \| = \| P_{\Omega, x_k} x_k - \nabla_x L(x_k, \lambda_k, \rho_k) - x_k \| \to 0$ then $v_k \to x^* \in int(\Omega)$. Then there exists $N \in \mathbb{N}$ such that $\forall k \geq N, \ lb < v_k < ub$. According to the exclusion condition in the system (74), $[\mu^*_i]_i = [\mu^*_j]_j = 0 \ \forall \ j$. Hence

$$\lim_{k \to \infty} \left[ \nabla_x L(x_k, \bar{\lambda}_k, \rho_k) \right] = 0 \quad (79)$$

There exists $\bar{\tau} \in \mathbb{R}^m_+$ and $\rho^* \in \mathbb{R}^m_+$ such that

$$\nabla_x L(x^*, \bar{\tau}, \rho^*) = 0 \quad (80)$$

Then $x^*$ is an optimal point and the Lagrange multiplicateur associated is $\bar{\tau}$.

**Consequence 3.1. (Convergence to an optimal point)**

Assume that the sequence $(x_k)_k$ is generated by algorithm C2 and that a limit point $x^*$ and that there exists $k_0 \geq 0$ such that $\forall k \geq k_0, \Gamma_k$ is empty set. Then $x^*$ is a KKT point.

**Proof**

According to the proposition (3.3), $x^*$ is feasible and $g(x^*) = 0$. Apply the proposition (3.1) we have $\lambda^* \in \mathbb{R}^m_+$ and $\rho^* \in \mathbb{R}^m_+$ such that

$$\nabla_x L(x^*, \lambda^*, \rho^*) = \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i [g^*_i](x^*) \cdot \nabla g_i(x^*) \quad (81)$$

$$= \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) + \sum_{i \in I} [\rho^*_i]_i [g^*_i](x^*) \cdot \nabla g_i(x^*) \quad (82)$$

$$= \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) \quad (83)$$

According to (80), we can conclude tha $x^*$ is a KKT point and the Lagrange multiplicateur associated is $\lambda^*$.

**Consequence 3.2. (Convergence to KKT point)**

Assume that the sequence $(x_k)_k$ is generated by algorithm C1 or C2 and that a limit point $x^*$ is feasible and the familly $(\nabla g_i(x^*))_{i \in I(x^*)}$ is positively linearly dependent with the coefficient $\alpha_i = -\rho^*_i g_i(x^*) \ \forall \ i \in I \setminus I(x^*)$. Then $x^*$ is KKT point.

**Proof**

It is easy to see that

$$\nabla_x L(x^*, \lambda^*, \rho^*) = \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) \quad (84)$$

$$= \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) + \sum_{i \in I} [\rho^*_i]_i [g^*_i](x^*) \cdot \nabla g_i(x^*) \quad (85)$$

$$= \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) + \sum_{i \in I \setminus I(x^*)} [\rho^*_i]_i [g^*_i](x^*) \cdot \nabla g_i(x^*) \quad (86)$$

$$= \nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) - \sum_{i \in I \setminus I(x^*)} [\rho^*_i]_i [g^*_i](x^*) \cdot \nabla g_i(x^*) \quad (87)$$

$$\forall i \in I \setminus I(x^*), \ -g_i(x^*) > 0, \ as \ \sum_{i \in I \setminus I(x^*)} [\rho^*_i]_i [g^*_i](x^*) \cdot \nabla g_i(x^*) = 0 \ because \ (\nabla g_i(x^*))_{i \in I(x^*)} \ is \ positively \ linearly \ dependent \ with \ the \ coefficient \ \alpha_i = -\rho^*_i g_i(x^*) \ \forall \ i \in I \setminus I(x^*), \ we \ have$$

$$\nabla f(x^*) + \sum_{i \in I} [\lambda^*_i]_i \cdot \nabla g_i(x^*) = \nabla_x L(x^*, \lambda^*, \rho^*) = 0 \quad (88)$$

Then $x^*$ is a KKT point.
3.3 Numerical Tests

We applied these algorithms to solve several problems of references (Hock, W. & Schittkowski, K., 1981; Asaadi, J., 1973; Miele, A. & et al., 1972; Miele, A. & et al., 1972; Biggs, M. C., 1971; Bracken J., 1976; Klaus, S., 2009), and results are presented in the following tables. For simulations, essentially with the algorithm \( C_2 \), let \( \tau = 10^{-5} \), \( \gamma = 10 \), \( \epsilon_{\text{opt}} = 10^{-8} \) and \( \epsilon_{f_{\text{es}}} = 10^{-8} \). The numerical tests were performed by using the software Python (Python Software Foundation) on a computer: Intel(R) Core(TM)2 Duo CPU 2.60GHz, 8.0Gb of RAM, under UNIX system.

The difference between the two algorithms is in their ways of computations of Lagrange multipliers and penalty parameters. The algorithm \( C_1 \) is specifically designed to solve the large-scale problems.

The Figures 1 2 3 4 5 and 6 (at the end of the tables) represent the evolution of the optimal value as well as the norm of the lagrangian gradient. The behavior of the curves demonstrates the rapid convergence of the algorithm towards the optimum point. However, in the case of problem 5, although the exact solution has been obtained, the curves show us an insufficiency in the convergence that we must solve for the continuation of our work in this direction.

We define by: NIter: Number of iterations, MaxIter: Maximal number of iterations, Nv: Number of variable, Nc: Number of constraints, f(x): Objective function, Sol.Alg: the solution found by our algorithm, Sol.ex: Solution found by the source.

Table 1. Results of Problem 1

<table>
<thead>
<tr>
<th>Problem</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>PRB-TP37</td>
</tr>
<tr>
<td>Nv</td>
<td>3</td>
</tr>
<tr>
<td>Nc</td>
<td>2</td>
</tr>
<tr>
<td>( f(x) )</td>
<td>(-x_1 \cdot x_2 \cdot x_3)</td>
</tr>
</tbody>
</table>
| Constraints | \[
\begin{align*}
-x_1 + 2x_2 + 3x_3 &\leq 72 \\
-3x_1 - 2x_2 - 3x_3 &\leq 0 \\
0 \leq x \leq 42 &\quad \forall i = 1, 2, 3
\end{align*}
\] |
| Start point | \( x_0 = (10, 10, 10) \) |
| Stop criterions | \( \|x_{k+1} - x_k\| \leq \epsilon_{\text{opt}} \) or \( \|f(x_{k+1}) - f(x_k)\| \leq \epsilon_{\text{opt}} \) |
| NIter/MaxIter | 141/1000 |
| Sol.Alg | \((23.99999993, 12.00000002, 12.00000002)\) (feasible) |
| Sol.ex | \((24, 12, 12)\) |

Table 2. Results of Problem 2

<table>
<thead>
<tr>
<th>Problem</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>PRB-T1-3</td>
</tr>
<tr>
<td>Source</td>
<td>Asaadi, J.(Asaadi, J., 1973)</td>
</tr>
<tr>
<td>Nv</td>
<td>2</td>
</tr>
<tr>
<td>Nc</td>
<td>2</td>
</tr>
<tr>
<td>( f(x) )</td>
<td>(\frac{1}{4}(x_1 + 1)^2 + x_2)</td>
</tr>
</tbody>
</table>
| Constraints | \[
\begin{align*}
1 - x_1 &\leq 0 \\
-x_2 &\leq 0
\end{align*}
\] |
| Start point | \( x_0 = (1.125, 0.125) \) |
| Stop criterions | \( \|x_{k+1} - x_k\| \leq \epsilon_{\text{opt}} \) or \( \|f(x_{k+1}) - f(x_k)\| \leq \epsilon_{\text{opt}} \) |
| NIter/MaxIter | 4/1000 |
| Sol.Alg | \((1.00000000e + 00, 1.10201928e - 08)\) (feasible) |
| Sol.ex | \((1, 0)\) |
Table 3. Results of Problem 3

<table>
<thead>
<tr>
<th>Problem</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>GPR-T1-1</td>
</tr>
<tr>
<td>Source</td>
<td>Miele et al.(Miele, A. &amp; et al., 1972; Coggins, G. M. &amp; et al, 1972)</td>
</tr>
<tr>
<td>( N_v )</td>
<td>( n = 2 )</td>
</tr>
<tr>
<td>( N_c )</td>
<td>( m = 1 )</td>
</tr>
<tr>
<td>( f(x) )</td>
<td>( \log(x_1^2 + 1) - x_2 )</td>
</tr>
<tr>
<td>Constraints</td>
<td>((x_1^2 + 1)^2 + x_2^2 - 4 = 0)</td>
</tr>
<tr>
<td>Start point</td>
<td>( x_0 = (2, 2) )</td>
</tr>
<tr>
<td>Stop criteria</td>
<td>( |x_{k+1} - x_k| \leq \varepsilon_{\text{opt}} ) or ( |f(x_{k+1}) - f(x_k)| \leq \varepsilon_{\text{opt}} )</td>
</tr>
<tr>
<td>( N_{\text{Iter}}/\text{MaxIter} )</td>
<td>( 8/1000 )</td>
</tr>
<tr>
<td>Sol.Alg</td>
<td>((2.23896809e - 09, 1.73205081e + 00)) (feasible)</td>
</tr>
<tr>
<td>Sol.ex</td>
<td>((0, \sqrt{3})) (feasible)</td>
</tr>
</tbody>
</table>

Table 4. Results of Problem 4

<table>
<thead>
<tr>
<th>Problem</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>GLR-T1-1</td>
</tr>
<tr>
<td>Source</td>
<td>Miele et al.(Miele, A. &amp; et al., 1972)</td>
</tr>
<tr>
<td>( N_v )</td>
<td>( n = 2 )</td>
</tr>
<tr>
<td>( N_c )</td>
<td>( m = 1 )</td>
</tr>
<tr>
<td>( f(x) )</td>
<td>( \sin(\frac{x_1}{12}) \cdot \cos(\frac{x_2}{12}) )</td>
</tr>
<tr>
<td>Constraints</td>
<td>( 4x_1 - 3x_2 \leq 0 )</td>
</tr>
<tr>
<td>Start point</td>
<td>( x_0 = (0, 0) )</td>
</tr>
<tr>
<td>Stop criteria</td>
<td>( |x_{k+1} - x_k| \leq \varepsilon_{\text{opt}} ) or ( |f(x_{k+1}) - f(x_k)| \leq \varepsilon_{\text{opt}} )</td>
</tr>
<tr>
<td>( N_{\text{Iter}}/\text{MaxIter} )</td>
<td>( 34/1000 )</td>
</tr>
<tr>
<td>Sol.Alg</td>
<td>((-2.999998922, -3.99998563)) (feasible)</td>
</tr>
<tr>
<td>Sol.ex</td>
<td>((-3, -4))</td>
</tr>
</tbody>
</table>
Table 5. Results of Problem 5

<table>
<thead>
<tr>
<th>Problem</th>
<th>5 (cattel-feed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>LGI-P1-1</td>
</tr>
<tr>
<td>$N_v$</td>
<td>$n = 4$</td>
</tr>
<tr>
<td>$N_c$</td>
<td>$m = 3$</td>
</tr>
<tr>
<td>$f(x)$</td>
<td>$24.55x_1 + 26.75x_2 + 39x_3 + 40.5x_4$</td>
</tr>
</tbody>
</table>
| Constraints | $\begin{align*} x_1 + x_2 + x_3 + x_4 - 1 &= 0 \\
-2.3x_1 - 5.6x_2 - 11.1x_3 - 1.3x_4 + 5 &\leq 0 \\
-12x_1 - 11.9x_2 - 42.8x_3 - 51.4x_4 + 21 + 1.645[0.28x_1^3 + 0.19x_2^3 + 20.5x_3^2 + 0.62x_4^2] &\leq 0 \end{align*}$ |
| Start point | $x_0 = (1, 1, 1, 1)$ |
| Stop criterions | $\|x_{k+1} - x_k\| \leq \varepsilon_{opt} \text{ or } \|f(x_{k+1}) - f(x_k)\| \leq \varepsilon_{opt}$ |
| $N_{Iter}/MaxIter$ | 5/1000 |
| Sol.Alg | $(6.42805324e-01, 1.18335296e-08, 3.11958636e-01, 4.52360285e-02)$ (feasible) |
| Sol.ex | $(0.6355216, -0.12e - 11, -0.3127019, 0.05177655)$ (unfeasible) |

Table 6. Results of Problem 6

<table>
<thead>
<tr>
<th>Problem</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>PQR-T1-2</td>
</tr>
<tr>
<td>Source</td>
<td>Asaadi J. (Asaadi, J., 1973)</td>
</tr>
<tr>
<td>$N_v$</td>
<td>$n = 2$</td>
</tr>
<tr>
<td>$N_c$</td>
<td>$m = 2$</td>
</tr>
<tr>
<td>$f(x)$</td>
<td>$100(x_2 - x_1^2)^2 + (1 - x_1)^2$</td>
</tr>
</tbody>
</table>
| Constraints | $\begin{align*} x_1 + x_2^2 &\geq 0 \\
x_1^2 + x_2 &\geq 0 \\
-2 &\leq x_1 \leq 0.5, \quad x_2 \leq 1 \end{align*}$ |
| Start point | $x_0 = (0.1)$ |
| Stop criterions | $\|x_{k+1} - x_k\| \leq \varepsilon_{opt} \text{ or } \|f(x_{k+1}) - f(x_k)\| \leq \varepsilon_{opt}$ |
| $N_{Iter}/MaxIter$ | 11/1000 |
| Sol.Alg | $(0.5, 0.25)$ (feasible) |
| Sol.ex | $(0.5, 0.25)$ |
Table 7. Results of Problem 7

<table>
<thead>
<tr>
<th>Problem</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>PQ-T1-1</td>
</tr>
<tr>
<td>( N_v )</td>
<td>( n = 13 )</td>
</tr>
<tr>
<td>( N_c )</td>
<td>( m = 9 )</td>
</tr>
</tbody>
</table>
| \( f(x) \) | \[
5 \left( \sum_{i=1}^{4} x_i \right) - 5 \left( \sum_{i=1}^{4} x_i^2 \right) - \sum_{i=5}^{13} x_i
\]
| Constraints | \[
\begin{align*}
2x_1 + x_2 + x_{10} + x_{11} - 10 & \leq 0 \\
2x_1 + 2x_3 + x_{10} + x_{12} - 10 & \leq 0 \\
2x_2 + 2x_3 + x_{11} + x_{12} - 10 & \leq 0 \\
-8x_1 + x_{10} & \leq 0 \\
-8x_2 + x_{11} & \leq 0 \\
-8x_3 + x_{12} & \leq 0 \\
-2x_4 - x_5 + x_{10} & \leq 0 \\
-2x_6 - x_7 + x_{11} & \leq 0 \\
-2x_8 - x_9 + x_{12} & \leq 0
\end{align*}
\]
| Start point | \( x_0 = \cos(\text{ones}(13)) \) |
| Stop criterions | \( ||x_{k+1} - x_k|| \leq \varepsilon_{\text{opt}} \) or \( ||f(x_{k+1}) - f(x_k)|| \leq \varepsilon_{\text{opt}} \) |
| \( N_{\text{Iter}}/\text{MaxIter} \) | 11/1000 |
| Sol.Alg | \((1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 3, 0.99999999)\) (feasible) |
| Sol.ex | \((1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3)\) |

Figure 1. Evolution of the optimal value and the norm of the Lagrangian gradient
Figure 2. Evolution of the optimal value and the norm of the Lagrangian gradient

Figure 3. Evolution of the optimal value and the norm of the Lagrangian gradient
Figure 4. Evolution of the optimal value and the norm of the Lagrangian gradient

Figure 5. Evolution of the optimal value and the norm of the Lagrangian gradient
Figure 6. Evolution of the optimal value and the norm of the Lagrangian gradient

Figure 7. Evolution of the optimal value and the norm of the Lagrangian gradient
4. Conclusion

In this work, we proposed an algorithm for solving optimization problems under inequality constraints. We obtained convergence of generated sequence to an optimal solution, satisfying the Karush-Kuhn-Tucker qualification constraints. Simulations on academic data have shown the performance of our method. Indeed, we made changes in the calculation of the multipliers for searching the point $x_{k+1}$ when $x_k$ is already determined; and defined new set $\Omega(x_k)$ associated to each point $x_k$ for the projection in order to promote rapid convergence to an feasible point.

Acknowledgement

The authors thank the anonymous referees for useful comments and suggestions. Gratitute is expressed to the project Centre d’Excellence Africain en Sciences Mathématiques et Applications (CEA-SMA) for the support of this work.

References


Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/4.0/).