Yugoslav Journal of Operations Research
xx (), Number nn, zzz–zzz
DOI: https://doi.org/10.2298/YJOR251015011C

Research Article

UNCONSTRAINED REFORMULATION OF SEQUENTIAL QUADRATIC PROGRAMMING FOR SOLVING OPTIMIZATION PROBLEMS WITH LINEAR EQUALITY CONSTRAINTS

Suvra Kanti CHAKRABORTY
Department of Mathematics, Ramakrishna Mission Vidyamandira, Howrah, India
suvrakanti@vidyamandira.ac.in, ORCID: 0000-0002-1245-7839

Mohammad Esmael SAMEI
Department of Mathematics, Faculty of Science, Bu-Ali Sina University, Hamedan, Iran
mesamei@basu.ac.ir, mesamei@gmail.com, ORCID: 0000-0002-5450-3127

Rudrajit SADHU*
Department of Mathematics, Sreegopal Banerjee College, Magra, India
rudrajitsadhu@gmail.com, ORCID: 0000-0002-1487-1750

Received: October 2025 / Accepted: June 2026

Abstract: In this paper, we propose an alternative approach to Sequential Quadratic Programming (SQP) for solving optimization problems with linear equality constraints. The concept of a regularized gap function is employed to reformulate the constrained problem into an unconstrained one. The proposed algorithm incorporates a positive definite Hessian modification along with a backtracking line search to promote global convergence. Its effectiveness is demonstrated through several numerical experiments and geometric illustrations. Furthermore, applications of the proposed method are explored in optimal allocation, analytic center computation, and network flow problems.

Keywords: Sequential quadratic programming, constrained optimization, unconstrained reformulation, convex optimization, local minimum point.

MSC: 90C26, 90C30.

*Corresponding author

1. INTRODUCTION

Optimization has become an indispensable tool across a wide range of scientific and engineering disciplines. Both its direct and indirect applications now play a prominent role in solving practical problems in areas such as industrial engineering, manufacturing systems, multi-criteria decision-making, information systems, neural networks, control theory, and matrix theory.

An optimization problem ($\mathbb{OP}$) involves the quest for the best solution to practical problems through mathematical formulation. An $\mathbb{OP}$, if it involves restrictions on these variables, can be categorized as a constrained optimization problem ($\mathbb{C}\mathbb{OP}$). We can have a look at a general nonlinear $\mathbb{C}\mathbb{OP}$, which is of the following form

$$\begin{cases} \quad\quad \min & \mathrm{v}(\mathfrak{s}), \\ \text{subject to} & h_{1_{\iota_1}}(\mathfrak{s}) \leq \xi_{\iota_1}\,(\iota_1 = 1,2,\ldots,\aleph_1), \\ & h_{2_{\iota_2}}(\mathfrak{s}) = \xi_{\iota_2}\,(\iota_2 = 1,2,\ldots,\aleph_2), \end{cases} \tag{1}$$

where v is a nonlinear function and the functions $h_{1_{\iota_1}}, h_{2_{\iota_2}} : \mathbb{R}^n \to \mathbb{R}$ are inequality and equality constraints respectively and ξ_{ι_1} and ξ_{ι_2}'s are real numbers. There are numerous successful numerical methods for solving the above nonlinear $\mathbb{C}\mathbb{OP}$. Interior & Exterior penalty function methods are some indirect methods. On the contrary, Sequential Linear Programming ($\mathbb{SLP}$), Active Set method ($\mathbb{ASM}$), and Sequential Quadratic Programming ($\mathbb{SQP}$) are some of the direct methods for capturing its solution. An indirect method works by analytically constructing the necessary and sufficient conditions for optimality, which are then solved numerically. A direct method differs from an indirect one in its attempt to find the numerical solution by constructing a sequence of continually improving approximations to the optimal solution. The theoretical and numerical illustrations for the above-mentioned techniques can be perused in [1].

Among all these methods, SQP is one of the most widely used and efficient techniques, which: (i) approximates the original constrained optimization problem by a sequence of quadratic programming (QP) subproblems; (ii) linearizes the constraints at the current iterate; (iii) solves the resulting QP subproblem to determine a search direction and generate a new iterate; and (iv) repeats the above process iteratively until convergence to a Karush–Kuhn–Tucker (KKT) point. Wilson [2] first proposed this technique in his dissertation in 1963. Since then, the $\mathbb{SQP}$ method drew mathematicians' and engineers' attention for its efficiency and simplicity in implementation. $\mathbb{SQP}$ along with its enormous variations are being studied extensively by many researchers [3, 4, 5, 6, 7]. The pursuers may see Boggs [8], Gould *et al.* [9, 10] for the chronological development and a good understanding of $\mathbb{SQP}$ algorithm. Problem (1) represents a general nonlinear constrained optimization framework involving both inequality and equality constraints. In the present work, however, we restrict our attention to optimization problems with equality constraints only, which takes the form;

$$\mathbb{ECP}: \quad \begin{cases} \quad\quad \min & \mathrm{v}(\mathfrak{s}) \\ \text{subject to} & h_{2_{\iota_2}}(\mathfrak{s}) = \xi_{\iota_2}\,(\iota_2 = 1,2,\ldots,\aleph_2), \end{cases}$$

This specialized setting enables the construction of an explicit projection operator and facilitates the unconstrained reformulation developed in the subsequent sections.

We now consider an equality constrained problem to provide an overview of the $\mathbb{SQP}$ method. At each iteration, the method solves a quadratic subproblem ($\mathbb{Q}\,\text{sub}\mathbb{P}$) of the given problem in matrix form as follows

$$\mathbb{Q}\,\text{sub}\mathbb{P}: \qquad \begin{cases} \qquad \min\limits_{\mathfrak{s}} & \hat{\mathrm{v}}(\mathfrak{s}) = \frac{1}{2}\,\mathfrak{s}^T H\mathfrak{s} + \mathrm{c}^T\mathfrak{s} \\ \text{subject to} & U\mathfrak{s} = \mathring{u}. \end{cases} \tag{2}$$

The $\mathbb{Q}\,\text{sub}\mathbb{P}$ (2) is the quadratic-linear approximation to the $\mathbb{ECP}$. It considers the quadratic approximation of the objective function and the linear approximation of the constraints. Hence the vector c signifies the gradient vector $\nabla\mathrm{v}$ of the objective function, and H denotes the Hessian of the objective function or an approximation to it.

The solution to $\mathbb{Q}\,\text{sub}\mathbb{P}$ (2) represents a search direction for the original problem. The linearization of the equality constraints at a current iterate of an optimization algorithm produces the system of a linear equation $U\mathfrak{s} = \mathring{u}$. An assumption on $U = [u_\iota]_{m_2\times m_1}$ ($m_2 < m_1$), is that U must possess full row rank i.e. the linear system has m_2 linearly independent equations and $\mathring{u} \in \mathbb{R}^{m_2}$. Another assumption on H is that H must be positive semi-definite on the null space of the constraint U, which theoretically guarantees that $\mathbb{Q}\,\text{sub}\mathbb{P}$ has a solution. The $\mathbb{Q}\,\text{sub}\mathbb{P}$ is associated with a KKT system, a $m_1 + m_2$ linear equation in $m_1 + m_2$ unknowns by first-order necessary optimality condition, which reduces the $\mathbb{QP}$ to the issue of solving a system of linear equations. A few factorizing procedures are accessible for solving the KKT system straightforwardly (see [6, 1]) or else distinctive iterative strategies (see [9]) can be utilized to solve the system up to a desired level of accuracy, these strategies suit well for huge systems.

In this paper, we concentrate on the ECPs with linear equality constraints. However, the objective function may be convex or non-convex. In other words, the Hessian matrix of the objective function may not be positive definite. In the case of the negative definite or indefinite Hessian matrix, we use a positive definite modification of it and construct an equivalent convex optimization problem. Instead of solving the KKT system related with the $\mathbb{Q}\,\text{sub}\mathbb{P}$, we propose an exact unconstrained reformulation (UR) of the $\mathbb{Q}\,\text{sub}\mathbb{P}$ using a projection map on the feasible set. The reformulation uses "regularized gap function", ($\mathbb{RGF}$), originated from variational inequality problem ($\mathbb{VIP}$), introduced by Fukushima [11]. Several varieties of gap function for variational inequalities are available in the literature [12, 13, 14]. One may find a detailed survey of the $\mathbb{GF}$ in [15]. In the course of its application, Li *et al.* first used the $\mathbb{RGF}$ to propose an exact penalty function for minimizing a twice continuously differentiable function over a convex set [16]. Li proved that both the original and the reformulated problems have the same set of local and global solutions under certain assumptions [16]. Though this theory's basis and results remain very strong, its implementation to a general class of function encounters several difficulties. This fact motivates us to investigate the theory for the simplest case, where the theory proposed in [16] is most applicable. Earlier studied some numerical behavior and extended the idea for solving a convex $\mathbb{OP}$ [17, 18].

This article develops a modified $\mathbb{SQP}$-type scheme based on an unconstrained reformulation and provides geometrical interpretations through contour plots and feasible-set projections to illustrate the behavior of the iteration sequence. The theory of positive definite approximation, the idea of projection, and the backtracking line-search strategy play vital roles in constructing the proposed scheme. Classical SQP methods typically compute the search direction by solving the KKT system associated with a quadratic programming subproblem at each iteration. In contrast, the proposed approach reformulates the subproblem into an equivalent unconstrained optimization problem, thereby avoiding the explicit solution of the KKT system.

The paper is organized as follows. In Section 2, we briefly discuss the prerequisites and mathematical backgrounds. In Section 3, the main scheme is proposed along with a proper algorithm. Several geometrical illustrations with suitable numerical examples and various applications of the proposed scheme are provided in Section 4, followed by some concluding remarks in Section 5.

2. PREREQUISITES AND MATHEMATICAL FOUNDATION

The following notations are used throughout the paper.

N1) For a matrix $U = [u_\iota]_{m_2 \times m_1}$, U^T denotes the transpose of U.

N2) The Euclidean norm$\|\mathfrak{s}\|$ is defined by

$$\|\mathfrak{s}\| = \left[\sum_{=1}^{n} \mathfrak{s}^2\right]^{\frac{1}{2}}, \qquad \mathfrak{s} = \left[\mathfrak{s}_1 \mathfrak{s}_2 \ldots \mathfrak{s}_n\right]^T.$$

N3) $\mathfrak{s}^k$ denotes the vector at k-th iteration,

N4) $v_k = v(s^k)$, while ∇v_k and $\nabla^2 v_k$ denote, respectively, the gradient and Hessian of v evaluated at s^k.

N5) $\|U\|$ denotes the usual operator norm.

N6) $\|U\|_{\mathrm{F}}$ denotes the Frobenius norm of the matrix U,

$$\|U\|_F = \left[\sum_{=1}^{m_1} \sum_{\iota=1}^{m_2} u_\iota^2\right]^{1/2}.$$

2.1. $\mathbb{GF}$ and reformulation

A $\mathbb{COP}$ for a twice continuously differentiable function $\mathrm{v} : \mathbb{R}^{m_1} \to \mathbb{R}$ is defined by

$$\min_{\mathfrak{s} \in S} \mathrm{v}(\mathfrak{s}), \tag{3}$$

where $S \subseteq \mathbb{R}^{m_1}$ is a closed convex set. We call a point $\bar{\mathfrak{s}}$ is a stationary point for the constrained minimization problem ($\mathbb{CMP}$) of $\mathbb{COP}$ if the variational inequality problem

$$\mathbb{VIP}: \qquad \bar{\mathfrak{s}} : \left\langle \nabla \mathrm{v}(\bar{\mathfrak{s}}), \mathfrak{t} - \bar{\mathfrak{s}} \right\rangle \geq 0, \quad \forall\, \mathfrak{t} \in S. \tag{4}$$

holds for $\bar{\mathfrak{s}}$. Several varieties of gap functions for variational inequalities have been proposed in the literature, each possessing different analytical and computational properties. In order to solve the $\mathbb{VIP}$ (4), Fukushima proposed the concept of regularized $\mathbb{GF}$ [11].

By this $\mathbb{GF}$, (4) can be reformulated to an $\mathbb{OP}$. Employing the idea of regularized $\mathbb{GF}$, Li proposed an exact *UR* of $\mathrm{C}\mathbb{OP}$ [16]. The $\mathbb{G}_\eta(\mathfrak{s})$, used by both authors is as follows

$$\begin{aligned}
\mathbb{G}_\eta(\mathfrak{s}) &= \max_{\mathfrak{t}\in S}\left\{(\mathfrak{s}-\mathfrak{t})^T\nabla \mathrm{v}(\mathfrak{s}) - \tfrac{1}{2\eta}\|\mathfrak{s}-\mathfrak{t}\|^2\right\}\\
&= (\mathfrak{s}-\mathbb{H}_\eta(\mathfrak{s}))^T\nabla \mathrm{v}(\mathfrak{s}) - \tfrac{1}{2\eta}\|\mathfrak{s}-\mathbb{H}_\eta(\mathfrak{s})\|^2\\
&= \tfrac{\eta}{2}\|\nabla \mathrm{v}(\mathfrak{s})\|^2 - \tfrac{1}{2\eta}\|(\mathbb{H}_\eta(\mathfrak{s})-\mathfrak{s})+\eta\nabla \mathrm{v}(\mathfrak{s})\|^2,
\end{aligned} \tag{5}$$

where $\eta > 0$ is a penalty parameter dependent on v and

$$\mathbb{H}_\eta(\mathfrak{s}) = \mathrm{Proj}_S(\mathfrak{s}-\eta\nabla \mathrm{v}(\mathfrak{s})) = \arg\max_{\mathfrak{t}\in S}\left\{(\mathfrak{s}-\mathfrak{t})^T\nabla \mathrm{v}(\mathfrak{s}) - \tfrac{1}{2\eta}\|\mathfrak{s}-\mathfrak{t}\|^2\right\},$$

where $\mathrm{Proj}_S(\mathfrak{s})$ denotes the unique orthogonal projection of $\mathfrak{s}$ onto the closed convex set S. The unconstrained reformulation (UR) of $\mathrm{C}\mathbb{OP}$ as proposed is

$$\min_{\mathfrak{s}\in\mathbb{R}^{m_1}} \quad \mathbb{P}_\eta(\mathfrak{s}), \tag{6}$$

where $\mathbb{P}_\eta(\mathfrak{s}) = \mathrm{v}(\mathfrak{s}) - \mathbb{G}_\eta(\mathfrak{s})$ [16]. From (5), $\mathbb{P}_\eta(\mathfrak{s})$ reduces to

$$\mathbb{P}_\eta(\mathfrak{s}) = \mathrm{v}(\mathfrak{s}) + (\mathbb{H}_\eta(\mathfrak{s})-\mathfrak{s})^T\nabla \mathrm{v}(\mathfrak{s}) + \tfrac{1}{2\eta}\|\mathfrak{s}-\mathbb{H}_\eta(\mathfrak{s})\|^2. \tag{7}$$

It is interesting to note that $\mathbb{P}_\eta(\mathfrak{s})$ thus obtained is differentiable and its gradient is given by the following lemma.

Lemma 1 ([16]). *Suppose that $\mathbb{P}_\eta(\mathfrak{s})$ is defined as in* (7) *and* $\mathrm{v}(\mathfrak{s})$ *is twice continuously differentiable then*

$$\nabla\mathbb{P}_\eta(\mathfrak{s}) = \tfrac{1}{\eta}\left(I-\eta\nabla^2 \mathrm{v}(\mathfrak{s})\right)\left(\mathfrak{s}-H_\eta(\mathfrak{s})\right).$$

The expression for the gradient of $\mathbb{P}_\eta(\mathfrak{s})$ shows that any fixed point of $\mathbb{H}_\eta(\mathfrak{s})$ is always a stationary point of $\mathbb{P}_\eta(\mathfrak{s})$. The following characterization of $\mathbb{H}_\eta(\mathfrak{s})$ motivates some fixed point algorithms for solving $\mathbb{VIP}$ and $\mathrm{C}\mathbb{OP}$.

Lemma 2 ([19]). *A vector $\bar{\mathfrak{s}} \in \mathbb{R}^{m_1}$ is a solution of $\mathbb{VIP}$, iff $\bar{\mathfrak{s}} = \mathbb{H}_\eta(\bar{\mathfrak{s}})$.*

The above characterization of the projection operator is classical and can be found in standard references on nonlinear optimization; see, for example, [20]. Lemmas 1 and 2 yield any stationary point of problem $\mathrm{C}\mathbb{OP}$ is also a stationary point of $\mathbb{P}_\eta(\mathfrak{s})$. The equivalence of the original constrained minimization problem $\mathrm{C}\mathbb{OP}$ and the *UR* of $\mathbb{P}_\eta(\mathfrak{s})$ on $\mathbb{R}^{m_1}$ can be vividly presented in the following theorem.

Theorem 3 ([16]). *Consider the vector $\bar{\mathfrak{s}}$. The following statements hold.*

i) *Suppose* $\eta\|\nabla^2 \mathrm{v}(\bar{\mathfrak{s}})\| < 1$. *Then $\bar{\mathfrak{s}}$ is a local minimizer of* $\mathrm{v}(\mathfrak{s})$ *in S iff $\bar{\mathfrak{s}}$ is a local minimizer of* $\mathbb{P}_\eta(\mathfrak{s}) \in \mathbb{R}^{m_1}$*;*

ii) *Suppose* $\eta\|\nabla^2 \mathrm{v}(\mathfrak{s})\| < 1, \forall \mathfrak{s} \in \mathbb{R}^{m_1}$. *Then $\bar{\mathfrak{s}}$ is a global minimizer of* $\mathrm{v}(\mathfrak{s}) \in S$ *iff $\bar{\mathfrak{s}}$ is a global minimizer of* $\mathbb{P}_\eta(\mathfrak{s}) \in \mathbb{R}^{m_1}$.

The theorem says that by proper choice of the penalty parameter η, the unconstrained minimizer of the penalized objective function over the whole of $\mathbb{R}^{m_1}$ will solve the original problem. Thus, under the suitable assumption, every local and global minimizer of $\mathbb{P}_\eta(\mathfrak{s})$ is the local and global minimizer of the original objective function over the feasible set S. Despite these strong theoretical results, the exact penalty function proposed above has two major drawbacks.

- The penalty parameter η can be chosen explicitly, only when the norm of the Hessian matrix is bounded on the whole of $\mathbb{R}^{m_1}$.
- The construction of the penalty function demands the formula for the orthogonal projection of a vector on the set S, i.e., the explicit form of $\mathbb{H}_\eta(\mathfrak{s})$ is needed.

2.2. Positive Definite Modification

A real symmetric matrix $U = [u_\iota]_{m_1 \times m_1}$ can be factorized as

$$PUP^T = L\acute{U}L^T,$$

where L, P, and $\acute{U}$ denote a lower triangular matrix, a permutation matrix, and a block diagonal matrix, respectively, where $\acute{U}$ contains at most 2×2 blocks [21, 22, 23]. This symmetric indefinite factorization is implemented in MATLAB through the built-in command `ldl()`. Positive definite modifications of Hessian approximations are well studied in the SQP literature, including approaches based on secant updates and regularization techniques; see, for example, [24, 25, 26].

To modify U into a positive definite matrix, a correction factor τ_i is introduced as

$$\tau_i = \begin{cases} 0, & \lambda_i \geq \delta, \\ \delta - \lambda_i, & \lambda_i < \delta, \end{cases} \tag{8}$$

where λ_i denotes the eigenvalues of the block diagonal matrix, and δ is a small positive constant.

Algorithm 1 Modification of a Symmetric Indefinite Matrix to a Positive Definite One [1].

Require: Compute the factorization of the input matrix A as $PAP^T = LBL^T$.
1: Perform the spectral decomposition of B as $B = Q\Lambda Q^T$.
2: Construct $F = Q\,\mathrm{diag}(\tau_i)\,Q^T$, where τ_i is given by (8). Then $L(B+F)L^T$ becomes positive definite.
3: Since $P(A+E)P^T = L(B+F)L^T$, it follows that $E = P^T LFL^T P$.
Ensure: $\bar{A} \triangleq A + E$.

Through this process, $\bar{U} \triangleq U + E$ becomes a positive definite matrix. This procedure, described in [1], is efficiently implemented in MATLAB using the `ldl()` command, as it is computationally less expensive.

Remark: The positive definite modification used in Algorithm 1 is related in spirit to the Levenberg–Marquardt regularization technique, since both approaches aim to stabilize an indefinite or ill-conditioned Hessian approximation by enforcing positive definiteness. However, the present approach differs from the classical Levenberg–Marquardt method in an essential way. In the Levenberg–Marquardt framework, the modification is typically obtained by adding a scalar multiple of the identity matrix, i.e., $H+\lambda I$, where $\lambda > 0$ is a damping parameter. In contrast, Algorithm 1 employs a spectral correction based on the symmetric indefinite factorization of the matrix and modifies only those eigenvalues that fall below a prescribed threshold. Consequently, the correction is adaptive to the spectral structure of the matrix and generally produces a smaller perturbation than a uniform diagonal shift.

2.3. Armijo-Backtracking Line Search

Once a line search algorithm determines the search direction d_k for the next iteration, the selection of an appropriate step length becomes crucial. The Armijo-Backtracking line search is a practical technique that ensures a sufficient decrease in the objective function while preventing the step length from becoming excessively small. The step length β_k is determined according to the following procedure.

Algorithm 2 Armijo-Backtracking Line Search [1].

Require: Choose an initial step length $\beta_{\text{initial}} = 1$ and fix constants $c_1, \tau \in (0,1)$.
1: Set $k=0$ and $\beta_0 = \beta_{\text{initial}}$.
2: **while** $f(\mathfrak{s}^k + \beta_k d_k) \geq f(\mathfrak{s}^k) + c_1 \beta_k \nabla f(\mathfrak{s}^k)^T d_k$
3: $\quad \beta_{k+1} = \tau \beta_k$
4: $\quad k \leftarrow k+1$
Ensure: Final step length β_k.

3. UNCONSTRAINED REFORMULATION OF ECP

We consider a constrained optimization problem with linear equality constraints $(\mathbb{ECP}_l)$. This section develops the theory of the unconstrained reformulation (UR) of such problems and introduces a suitable algorithm to capture its local minimizer.

Consider

$$\mathbb{ECP}_l : \quad \begin{cases} \min\limits_{\mathfrak{s}} & \mathrm{v}(\mathfrak{s}) \\ \text{subject to} & U\mathfrak{s} = \mathring{u}, \end{cases} \tag{9}$$

where the function v is twice continuously differentiable, $U \in \mathbb{R}^{m_2 \times m_1}$ $(m_2 < m_1)$, and $\mathring{u} \in \mathbb{R}^{m_2}$. In particular, we assume throughout that U has full row rank, i.e.,

$$\operatorname{rank}(U) = m_2,$$

which guarantees the invertibility of UU^T and ensures the validity of the projection formula used in the sequel.

The motivation behind the $\mathbb{SQP}$ approach is to approximate $\mathbb{ECP}_l$ at a current iterate $\mathfrak{s}^k$ by a quadratic programming subproblem and then use its minimizer as a descent direction to generate the next iterate $\mathfrak{s}^{k+1}$. The approximate model of $\mathbb{ECP}_l$ at $\mathfrak{s}^k$ is

$$\mathbb{QP}_k : \quad \begin{cases} \min\limits_{\mathfrak{d}} \quad \bar{\mathrm{v}}_k(\mathfrak{d}) := \frac{1}{2}\mathfrak{d}^T \nabla^2 \mathrm{v}_k \mathfrak{d} + \nabla \mathrm{v}_k^T \mathfrak{d} + \mathrm{v}_k, \\ \text{subject to} \quad U\mathfrak{d} = \mathring{u} - U\mathfrak{s}^k. \end{cases} \tag{10}$$

In classical SQP frameworks, the Hessian of the Lagrangian is commonly used in the quadratic subproblem. However, in many practical implementations and modern SQP-type methods, approximations based on the Hessian of the primal objective function are also widely adopted. Since the constraints in ECPl are linear, they do not contribute to the curvature of the Lagrangian. Therefore, in this work, we employ the Hessian of the objective function, $\nabla^2 v_k$, as the basis for constructing the quadratic model. As the objective function may be non-convex, $\nabla^2 v_k$ is not necessarily positive definite. In such cases, we apply a positive definite modification to ensure that the resulting quadratic subproblem is convex and well-defined. Denote the modified matrix by Q_k.

With this modification, the convex quadratic approximation of $\mathbb{ECP}_l$ at $\mathfrak{s}^k$ is given by

$$\mathrm{C}\mathbb{QP}_k : \quad \begin{cases} \min\limits_{\mathfrak{d}} \quad q_k(\mathfrak{d}) := \frac{1}{2}\mathfrak{d}^T Q_k \mathfrak{d} + \nabla \mathrm{v}_k^T \mathfrak{d} + \mathrm{v}_k, \\ \text{subject to} \quad U\mathfrak{d} = \mathring{u} - U\mathfrak{s}^k. \end{cases} \tag{11}$$

The KKT optimality conditions of (11), where λ denotes the Lagrange multiplier corresponding to the equality constraints, are

$$Q_k \mathfrak{d} + \nabla \mathrm{v}_k - U^T \lambda = 0, \tag{12a}$$

$$U\mathfrak{d} + U\mathfrak{s}^k - \mathring{u} = 0. \tag{12b}$$

The K–K–T system (12) represents $m_1 + m_2$ linear equations in $m_1 + m_2$ unknowns $(\mathfrak{d}; \lambda)$. In the classical Newton–$\mathbb{SQP}$ method, this system is solved at each iteration to obtain $(\mathfrak{d}_k; \lambda_k)$, which then defines the next iterate $\mathfrak{s}^{k+1}$.

In contrast, instead of solving the K–K–T system (12) directly, we propose an unconstrained reformulation (UR) using the framework discussed in Subsection 2.1. The penalized objective function for the reformulated problem is

$$\mathbb{P}^k_\eta(\mathfrak{d}) = q_k(\mathfrak{d}) - \mathbb{G}^k_\eta(\mathfrak{d}). \tag{13}$$

For simplicity, let $C_k = \nabla \mathrm{v}_k$ and $\mathring{u}_k = \mathring{u} - U\mathfrak{s}^k$. As described in Subsection 2.1, constructing $\mathbb{P}^k_\eta(\mathfrak{d})$ requires the explicit expression for orthogonal projection onto the affine subspace S, and a suitable estimate for the penalty parameter η.

The orthogonal projection of any point $z \in \mathbb{R}^{m_1}$ onto the affine subspace

$$S = \{d \in \mathbb{R}^{m_1} : Ud = \mathring{u}_k\}$$

is

$$\mathrm{Proj}_S(z) = \left(I_{m_1} - U^T(UU^T)^{-1}U\right)z + U^T(UU^T)^{-1}\mathring{u}_k,$$

which is well defined since UU^T is invertible whenever U has full row rank.

Using this in (5), we obtain

$$\begin{aligned}\mathbb{G}^k_\eta(\mathfrak{d}) &= \tfrac{\eta}{2}\|Q_k\mathfrak{d} + C_k\|^2 \\ &\quad - \tfrac{1}{2\eta}\left\|U^T(UU^T)^{-1}\mathring{u}_k - U^T(UU^T)^{-1}U\left[(I_{m_1} - \eta Q_k)\mathfrak{d} - \eta C_k\right]\right\|^2.\end{aligned}$$

Substituting this expression into (13), we obtain

$$\mathbb{P}^k_\eta(\mathfrak{d}) = \mathfrak{d}^T R_k \mathfrak{d} + \mathrm{x}_k^T\mathfrak{d} + \mathrm{y}_k,$$

where $\bar{U} = U^T(UU^T)^{-1}U$, and

$$R_k = \tfrac{1}{2\eta}\left[I_{m_1} - (I_{m_1} - \eta Q_k)(I_{m_1} - \bar{U})\right](I_{m_1} - \eta Q_k),$$

$$\mathrm{x}_k = \left[C_k^T(I_{m_1} - \bar{U}) - \tfrac{1}{\eta}\mathring{u}_k^T(UU^T)^{-1}U\right](I_{m_1} - \eta Q_k),$$

$$\begin{aligned}\mathrm{y}_k &= \tfrac{1}{2\eta}\mathring{u}_k^T(UU^T)^{-1}\mathring{u}_k - \tfrac{\eta}{2}C_k^T(I_{m_1} - \bar{U})C_k \\ &\quad + C_k^T U^T(UU^T)^{-1}\mathring{u}_k + \mathrm{v}_k.\end{aligned}$$

Thus, the corresponding unconstrained reformulation of $(\mathrm{C}\mathbb{QP}_k)$ is

$$\mathrm{UC}\mathbb{QP}_k: \quad \min_{\mathfrak{d}\in\mathbb{R}^{m_1}} \mathbb{P}^k_\eta(\mathfrak{d}).$$

The function $\mathbb{P}^k_\eta(\mathfrak{d})$ is quadratic in $\mathfrak{d}$ and hence twice continuously differentiable for any $\eta > 0$. Since Q_k is positive definite and has bounded Frobenius norm, the penalty parameter η can be chosen such that

$$0 < \eta\|Q_k\|_F < 1,$$

which satisfies the sufficient condition required for exactness in Theorem 3. Since $\|Q_k\| \leq \|Q_k\|_F$, the condition $0 < \eta\|Q_k\|_F < 1$ implies $0 < \eta\|Q_k\| < 1$.

Consequently, under the above assumptions, the reformulated problem $\mathrm{UC}\mathbb{QP}_k$ admits a minimizer $\mathfrak{d}_k$. Moreover, the minimizer can be computed directly by solving the linear system

$$2R_k\mathfrak{d}_k + \mathrm{x}_k = 0. \tag{14}$$

The next iterate is computed via

$$\mathfrak{s}^{\bar{k}} = \mathfrak{s}^k + \mathfrak{d}_k,$$

followed by a backtracking line search (see Subsection 2.3) to ensure the descent property:

$$\mathfrak{s}^{k+1} = \mathfrak{s}^k + \beta_k\mathfrak{d}_k.$$

The iterations continue until an appropriate stopping criterion is satisfied.

Theorem 4. *If* $0 < \eta\|Q_k\|_F < 1$, *then* $\mathfrak{d}_k$ *minimizes*

$$q_k(\mathfrak{d}) = \frac{1}{2}\mathfrak{d}^T Q_k \mathfrak{d} + C_k^T \mathfrak{d} + \mathrm{v}_k$$

over S *if and only if* $\mathfrak{d}_k$ *minimizes* $\mathbb{P}_\eta^k(\mathfrak{d})$ *over* $\mathbb{R}^n$.

Proof. See [17, Theorem 3.1] for details. □

Advantages of the Scheme:

- The KKT system (12) associated with $\mathbb{ECP}_l$ consists of $m_1 + m_2$ linear equations in $m_1 + m_2$ unknowns, thereby increasing the computational dimension of the search process. In contrast, the reformulated problem $\mathrm{UC}\mathbb{QP}_k$ reduces the computation to an unconstrained minimization problem while implicitly preserving the equality constraints.
- The principal computational cost in constructing $\mathbb{P}_\eta^k$ arises from the inversion of an $m_2 \times m_2$ matrix. Hence, when $m_2 \ll m_1$, the reformulation becomes computationally efficient. This is particularly advantageous in resource allocation problems (see Subsection 4.1), where typically $m_2 = 1$ while the number of variables is large.
- The algorithm may be initialized from an arbitrary point, which is first projected onto the feasible set. Thereafter, feasibility is preserved throughout all iterations.
- The descent property is enforced through the incorporation of an Armijo backtracking line-search strategy.

Algorithm 3 Algorithm for Solving $\mathbb{ECP}_l$ via Unconstrained Reformulation

---

Require: Select an initial point $\mathfrak{s}^0 \in \mathbb{R}^{m_1}$ and a tolerance $\varepsilon > 0$.

Set $\mathfrak{s}^0 = \mathrm{Proj}_S(\mathfrak{s}^0)$.

repeat

  Evaluate v_k, $\nabla \mathrm{v}_k$, and $\nabla^2 \mathrm{v}_k$.

  Obtain Q_k as a positive definite modification of $\nabla^2 \mathrm{v}_k$ using Algorithm 1.

  Set

$$\eta_k = \frac{1}{1 + \|Q_k\|_F}.$$

  Evaluate R_k, x_k, and y_k using the relations derived in Section 3.

  Solve $(\mathrm{UC}\mathbb{QP}_k)$, i.e. the system of linear equation $2R_k\mathfrak{d}_k + \mathrm{x}_k = 0$, to obtain the search direction $\mathfrak{d}_k$.

  Determine the step length β_k using Algorithm 2.

  Update $\mathfrak{s}^{k+1} \leftarrow \mathfrak{s}^k + \beta_k \mathfrak{d}_k$.

until $\|\mathfrak{d}_k\| < \varepsilon$

Ensure: Approximate solution $\mathfrak{s}_{k+1}$.

Since the initial point $\mathfrak{s}^0$ is projected onto the feasible set, we have $U\mathfrak{s}^0 = \mathring{u}$. Also, from (11), if $\mathfrak{s}^k$ is feasible, then $U\mathfrak{d}_k = \mathring{u} - U\mathfrak{s}^k$. Since $U\mathfrak{s}^k = \mathring{u}$, it follows that $U\mathfrak{d}_k = 0$. Using the linearity of matrix multiplication, we obtain

$$U\mathfrak{s}^{k+1} = U(\mathfrak{s}^k + \beta_k \mathfrak{d}_k) = U\mathfrak{s}^k + \beta_k U\mathfrak{d}_k = U\mathfrak{s}^k = \mathring{u}.$$

Therefore, feasibility is preserved throughout the iterations.

Remark on convergence behavior: Since the matrix Q_k is constructed to be positive definite, the quadratic subproblem defining the search direction remains strictly convex at each iteration. Consequently, the computed direction $\mathfrak{d}_k$ is a descent direction for the reformulated objective function. Furthermore, the Armijo backtracking line-search strategy guarantees sufficient decrease of the objective function along the generated sequence. Therefore, under the assumptions that the objective function is continuously differentiable and bounded below on the feasible set, the generated sequence satisfies the standard descent properties associated with SQP-type line-search methods.

4. NUMERICAL ILLUSTRATIONS

We focus on the numerical aspect of our algorithm. First, we try to analyze the iteration scheme geometrically and then execute the proposed algorithm on a set of twelve linearly constrained test problems. The list includes convex as well as non-convex objective functions. To present a brief overview of the method, we consider the objective function $O_1(\mathfrak{s})$ subject to two different constraints $C_1(\mathfrak{s}) = 0$ and $C_2(\mathfrak{s}) = 0$ as follows.

$$O_1(\mathfrak{s}) = \mathfrak{s}_1^2 \sin \mathfrak{s}_1 - \mathfrak{s}_2^2 \cos \mathfrak{s}_2.$$

Constraints

$$C_1(\mathfrak{s}) \equiv \mathfrak{s}_1 - \mathfrak{s}_2 = 0, \quad C_2(\mathfrak{s}) \equiv \mathfrak{s}_1 + \mathfrak{s}_2 - 1 = 0.$$

For the geometrical interpretation, we formulate two problems; one $(\mathbb{P}1)$ and another $(\mathbb{P}2)$ of Table 1, where we minimize the objective function $O_1(\mathfrak{s})$ subject to the constraints $C_1(\mathfrak{s}) = 0$ and $C_2(\mathfrak{s}) = 0$ respectively. Figure 1 plots the three-dimensional surface of the objective function $O_1(\mathfrak{s})$ and Figures 2a and 2b present a combined contour plot of the objective function as well as the constraint sets. In Figure 2, the green curves represent the contour of the objective function, the black line plots the constraint set, and the red curve corresponds to the objective contour for the optimal value. The iteration points are also shown in the figure by small circles.

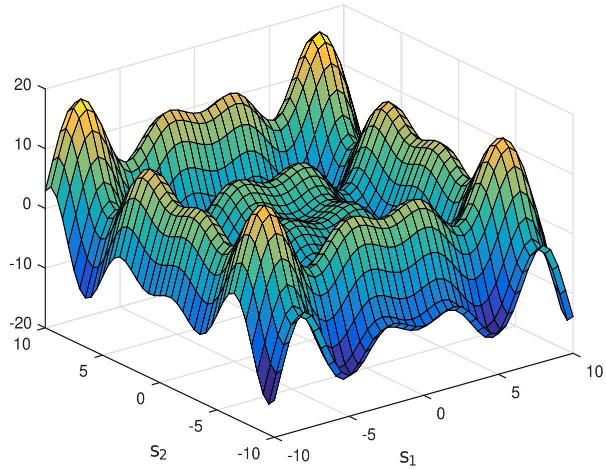
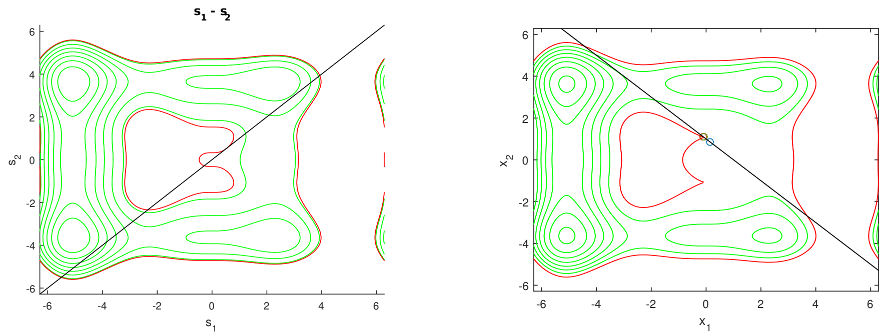


Figure 1: Three-dimensional plot of $O_1(\mathfrak{s})$



(a) subject to $C_1(\mathfrak{s}) = 0$

(b) subject to $C_2(\mathfrak{s}) = 0$

Figure 2: Minimization of $O_1(\mathfrak{s})$ subject to $C_1(\mathfrak{s}) = 0$ and $C_2(\mathfrak{s}) = 0$, respectively

Table 2 combines the results for executing the proposed Algorithm 3 on twelve constrained optimization problems whose objective functions are selected from [27]. In the said table, the constraints are written in the matrix form $U\mathfrak{s} = \mathring{u}$, where U is the matrix for the constraint set, and $\mathring{u}$ represents the constraint vector. $\mathrm{v}_{\min}$ denotes the functional value at the minimizer, and It_c denotes the iteration count to capture the solution of the problem. $\mathrm{rand}(p,q)$ defines a random matrix of order $p \times q$.

Table 1: Test Problems

Problem	Objective Function
$\mathbb{P}1$	$\mathfrak{s}_1^2 \sin \mathfrak{s}_1 - \mathfrak{s}_2^2 \cos \mathfrak{s}_2$
$\mathbb{P}2$	$\mathfrak{s}_1^2 \sin \mathfrak{s}_1 - \mathfrak{s}_2^2 \cos \mathfrak{s}_2$
$\mathbb{P}3$	$\mathfrak{s}_1^2 - \mathfrak{s}_2^2 + \mathfrak{s}_3^2 \log(1 + \mathfrak{s}_1^2 + \mathfrak{s}_2^2 + \mathfrak{s}_3^2)$
$\mathbb{P}4$	$5\sum_{i=1}^{13} \mathfrak{s}_i + 5\sum_{i=1}^{13} \mathfrak{s}_i^2 - \sum_{i=5}^{13} \mathfrak{s}_i$
$\mathbb{P}5$	$\mathfrak{s}_1^4 + \mathfrak{s}_2^4 + \mathfrak{s}_3^4 - \mathfrak{s}_1 - \mathfrak{s}_2 - \mathfrak{s}_3 + 1$
$\mathbb{P}6$	$\mathfrak{s}_1^4 + \mathfrak{s}_2^4 + \mathfrak{s}_3^4 - \mathfrak{s}_1 - \mathfrak{s}_2 - \mathfrak{s}_3 + 1$
$\mathbb{P}7$	$\sin(\mathfrak{s}_1 + \mathfrak{s}_2) + (\mathfrak{s}_1 - \mathfrak{s}_2)^2 - 1.5\mathfrak{s}_1 + 2.5\mathfrak{s}_2 + 1$
$\mathbb{P}8$	$(4 - 2.1\mathfrak{s}_1^2 + \frac{\mathfrak{s}_1^4}{3})\mathfrak{s}_1^2 + \mathfrak{s}_1\mathfrak{s}_2 + (-4 + 4\mathfrak{s}_2^2)\mathfrak{s}_2^2$
$\mathbb{P}9$	$30 + \sum_{i=1}^{3}(\mathfrak{s}_i^2 - 10\cos(2\pi\mathfrak{s}_i))$
$\mathbb{P}10$	$\sum_{i=1}^{10} \frac{\mathfrak{s}_i^2}{4000} - \prod_{i=1}^{10} \cos(\frac{\mathfrak{s}_i}{\sqrt{i}}) + 1$
$\mathbb{P}11$	$-\frac{1+\cos(12(\mathfrak{s}_1^2+\mathfrak{s}_2^2))}{0.5(\mathfrak{s}_1^2+\mathfrak{s}_2^2)+2}$
$\mathbb{P}12$	$\sum_{i=1}^{m_1} \sum_{j=1}^{i} \mathfrak{s}_j^2$

Table 2: Numerical performance and iteration comparisons for the test problems

Problem	U	$\hat{u}$	Starting Point	$\mathrm{v}_{\min}$	It_c	It_{RS}	It_{GP}
$\mathbb{P}1$	$[1,-1]$	0	$[2.5,-2]^T$	-0.10037	4	11	5
$\mathbb{P}2$	$[1,1]$	1	$[2.5,2]^T$	-0.5503	4	8	10
$\mathbb{P}3$	$[0,1,0]$	0	$[5,-3,1]^T$	1.02×10^{-11}	17	26	84
$\mathbb{P}4$	$\mathrm{rand}(1,13)$	1	$[1,1,\dots,1]^T$	-12.2	1	5	8
$\mathbb{P}5$	$[1,-1,1]$	1	$[2,1,2]^T$	-0.30752	11	11	102
$\mathbb{P}6$	$\begin{bmatrix} 1 & 2 & 3 \\ -3 & -2 & -1 \end{bmatrix}$	$[1,1]^T$	$[-1,-1,-1]^T$	1.1250	1	2	1
$\mathbb{P}7$	$[10,-7]$	2	$[0,-1]^T$	-1.7679	3	7	15
$\mathbb{P}8$	$[1,4]$	-2	$[5,5]^T$	-0.013	7	10	56
$\mathbb{P}9$	$\begin{bmatrix} 59 & -35 & 23 \\ -1 & 0 & 19 \end{bmatrix}$	$[20,11]^T$	$[0,0,0]^T$	20.792	1	1	12
$\mathbb{P}10$	$\begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 1 \end{bmatrix}$	$[1,1]^T$	$[1,1,\dots,1]^T$	0.465	2	8	1000*
$\mathbb{P}11$	$[1.375,-3.967]$	3.3	$[1,1]^T$	-0.6195	2	4	16
$\mathbb{P}12$	$\begin{bmatrix} 1 & 0 & 2 & 0 & 3 \\ 0 & -4 & 0 & -5 & 0 \end{bmatrix}$	$[0.2,0.3]^T$	$[1,1,1,1,1]^T$	0.00925	1	2	9

For the execution of the numerical experiments, we used MATLAB-2017a on a computer with 4GB RAM, an Intel(R) Core(TM) i5-2450M processor, and a 64-bit operating system. In Tables 1 and 2, the objective functions are taken as nonlinear functions and the constraint functions as linear. Here we execute Algorithm 3 with suitable starting points and an error tolerance limit of 10^{-3}. It is to be mentioned that the first iteration point for each problem is considered as

$$\mathfrak{s}^0 = \text{Proj}_S(\mathfrak{s}),$$

where the vector $\mathfrak{s}$ is taken from the "Starting Point" column of Table 2. The proposed algorithm solves all twelve test problems within a small number of iterations. We observe that, for quadratic objective functions with linear constraints, the proposed method attains the solution in a single iteration. This behavior is consistent with the standard local properties of SQP- and Newton-type methods, since in this case the quadratic approximation coincides exactly with the original problem. We have checked the correctness of the solutions by comparing them with the solution obtained by the solver $fmincon$ in MATLAB-2017a software.

Moreover, we considered two benchmark approaches from the literature for comparison with Algorithm 3 under the same computational setup. The first one is the range space method (RS), which plays an important role in subspace SQP methods [28, 29]. The second one is the Gradient Projection Method (GP), which is naturally applicable here due to the availability of an explicit projection formula for the affine feasible set. For the GP implementation, we employed the projected gradient iteration

$$\mathfrak{s}^{k+1} = \text{Proj}_S\left(\mathfrak{s}^k - \alpha_k \nabla \text{v}(\mathfrak{s}^k)\right),$$

where the step size α_k was determined using Armijo backtracking with parameters $c_1 = 10^{-4}$ and $\tau = 0.5$. For all methods, we used the same projected initial points, stopping criterion $\|d_k\| < 10^{-3}$, and software environment. In Table 2, the columns It_c, It_{RS}, and It_{GP} denote the iteration counts for the proposed algorithm, the range space method, and the Gradient Projection Method, respectively. The numerical results indicate that the proposed method generally requires fewer iterations for several nonlinear test problems. In particular, for strongly nonlinear or nonconvex problems such as $\mathbb{P}5$ and $\mathbb{P}8$, the Gradient Projection Method required substantially more iterations. This improvement may be attributed to the incorporation of second-order curvature information through the quadratic approximation together with the positive definite Hessian modification.

It is worth mentioning that, although the starting points are often remote from the optimal solution, all iteration points remain feasible throughout the algorithm. This follows from the fact that the search directions belong to $\ker(U)$ and the initial iterate is projected onto the feasible set. The penalty parameter is updated dynamically at each iteration according to the strategy described in Algorithm 3.

To further assess the efficiency of the proposed scheme, we also considered the Dolan–Moré performance profile [30] based on the iteration counts reported in Table 2. The profile compares the proposed unconstrained reformulation SQP method with the range space SQP approach and the Gradient Projection Method over the collection of test problems. The resulting profile shows that the proposed method solves a larger fraction of problems within a smaller performance ratio, indicating competitive performance in terms of iteration efficiency, particularly for nonlinear and nonconvex problems. It should be noted that the present numerical study primarily focuses on iteration-count comparisons.

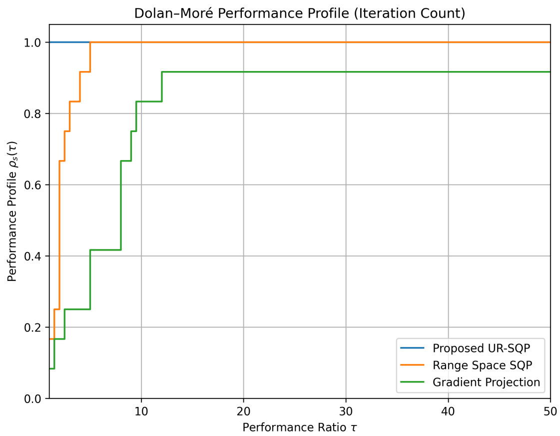


Figure 3: Dolan–Moré performance profile based on iteration counts

Remark: More extensive computational studies, including CPU-time analysis, runtime-based Dolan–Moré profiles, and comparisons with modern SQP variants, may be explored in future work. Nevertheless, the present results indicate that the proposed framework is effective for both convex and nonconvex equality constrained optimization problems.

4.1. Optimal allocation with resource constraints

An optimal allocation problem with resource constraints is defined as follows

$$\begin{cases} \min & \sum_{i=1}^{m_1} \mathrm{v}_i(\mathfrak{s}_i), \\ \text{subject to} & \sum_{i=1}^{m_1} \mathfrak{s}_i = \mathring{u}, \end{cases}$$

in which the functions v_i and $\mathring{u}$ are defined (2). This problem can be interpreted as allocating a single resource $\mathring{u}$ to the m_1 independent activities $\mathfrak{s}_i,\ i = 1, 2, \ldots, m_1$.

Example 5. *Let us consider an optimal network problem where*

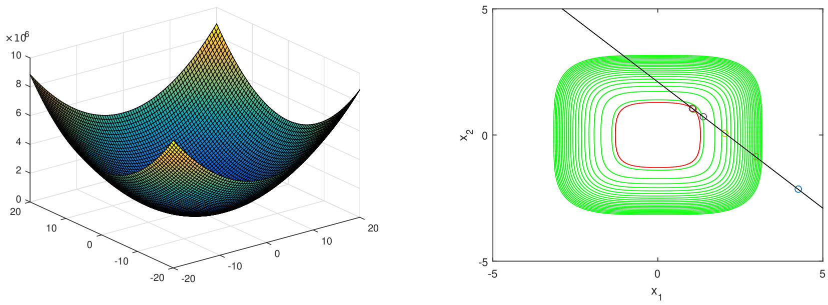
$$\mathrm{v}_i(\mathfrak{s}_i) = \mathfrak{s}_i^2 \log(1 + e^{\mathfrak{s}_i^2}), \qquad i = 1, 2, \ldots 15.$$

The constraint set is given by $U\mathfrak{s} = \mathring{u}$*, where* $U_{1\times 15} = [1, 1, \ldots, 1]$ *and* $\mathring{u} = 2.5$*. Now employing the proposed Algorithm 3 with starting vector*

$$[1, -1, 1, -1, \ldots, 1, -1, 1]^T,$$

we reach the minimizer of the optimization problem, i.e. $[0.1667, 0.1667, \ldots, 0.1667]$ *with minimum value* 0.29464 *in 5 iterations.*

Example 6. *If we adopt the two-dimensional version of the above optimization problem over the constraint* $\mathfrak{s}_1 + \mathfrak{s}_2 = 2.1$ *with initial vector* $[6, -4]^T$*, the minimum point* $[1.05, 1.05]$ *with functional value* 3.0632 *is achieved after six iterations. Figures 4a and 4b explain the three-dimensional plot of the objective function and the progression of iteration points side by side.*



(a) 3D view **(b)** 2D view

Figure 4: Three dimensional plot of objective function and progression of iteration points for Example 6

In both the above examples, the minimizers of the problems denote the optimal allocating units, and the objective functional value at the minimizer denotes the minimum possible cost for allocating such units.

4.2. Equality constrained analytic centering

An equality constrained analytic centering problem is defined as

$$\begin{cases} \min & -\sum_{i=1}^{m_1} \log \mathfrak{s}_i, \\ \text{subject to} & U\mathfrak{s} = \mathring{u}, \\ & \mathfrak{s}_i > 0, \quad i = 1, \dots, m_1. \end{cases}$$

where $U \in \mathbb{R}^{m_2 \times m_1}$ and $\mathring{u} \in \mathbb{R}^{m_2 \times 1}$. An analytic center is a point at which the normal forces associated with the linear constraints are balanced.

Example 7. *Let us consider the above constrained analytical centering problem in* 8 *variables. The constraint set is given by* $U\mathfrak{s} = \mathring{u}$*, where*

$$U = \begin{bmatrix} 0.6999 & 0.0688 & 0.6544 & 0.7184 & 0.3251 & 0.7788 & 0.2665 & 0.4401 \\ 0.6385 & 0.3196 & 0.4076 & 0.9686 & 0.1056 & 0.4235 & 0.1537 & 0.5271 \\ 0.0336 & 0.5309 & 0.8200 & 0.5313 & 0.6110 & 0.0908 & 0.2810 & 0.4574 \end{bmatrix}$$

and $\mathring{u} = [100, 105, 110]^T$*. Now employing the proposed Algorithm 3 with starting vector*

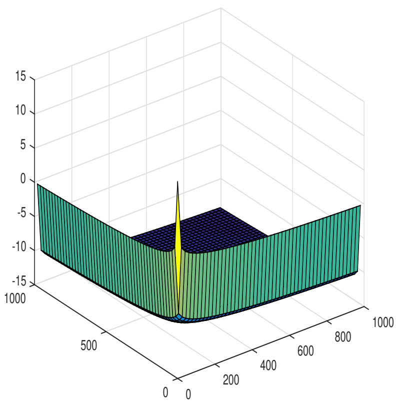
$$[10, 1, 1, 1, 1, 1, 1, 1]^T,$$

we reach the minimizer of the optimization problem, i.e.

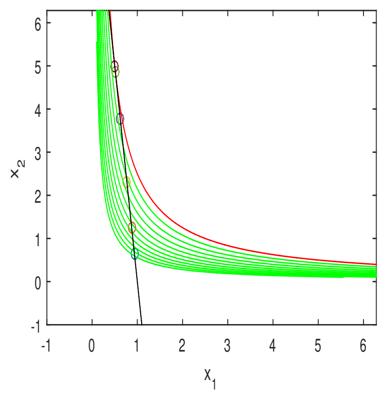
$$\left[27.885, 77.8, 15.865, 22.907, 24.854, 20.266, 41.04, 30.659\right],$$

with minimum value -26.937 *in* 8 *iterations.*

Example 8. *If we adopt the two-dimensional version of the above objective function and solve the optimization problem over the constraint* $3\mathfrak{s}_1 + 0.3\mathfrak{s}_2 = 3$*, with initial vector* $[0.0001, 0.235]^T$*, the minimum point* $[0.5, 4.9998]$ *with functional value* -0.91629*, is achieved after six iterations. Figures 5a and 5b explain the three-dimensional plot of the objective function and the progression of iteration points side by side.*



(a) 3D view



(b) 2D view

Figure 5: Three dimensional plot of the objective function and progression of iteration points for Example 8

In both the above examples, the minimizers of the problems denote the analytic centres, i.e., normal forces corresponding to the linear constraints get balanced at that point.

4.3. Optimal network flow

We will take traffic flows $\mathfrak{s}$ as the variables and the sources as given. We introduce the objective function as

$$\mathrm{v}(\mathfrak{s}) = \sum_{i=1}^{m_1} \phi_i(\mathfrak{s}_i),$$

where $\phi_i : \mathbb{R} \to \mathbb{R}$ is the flow cost function for arc $\imath$. We assume that the flow cost functions are strictly convex and twice differentiable.

The problem of choosing the best flow, that satisfies the flow conservation requirement, is defined as

$$\begin{cases} \quad \min & \displaystyle\sum_{i=1}^{m_1} \phi_i(\mathfrak{s}_i), \\ \text{subject to} & U\mathfrak{s} = \mathring{u}. \end{cases}$$

Since the objective function is separable, the Hessian matrix is diagonal.

Example 9. *Let us consider an optimal allocation problem where*

$$\phi_i(\mathfrak{s}_i) = \frac{\mathfrak{s}_i^4}{i}, \qquad i = 1, 2, \dots 10.$$

The constraint set is given by $U\mathfrak{s} = \mathring{u}$, where

$$U = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and $\mathring{u} = [-1, -1, -1, -1]^T$. Now employing the proposed Algorithm 3 with starting vector

$$\big[1, 1, 1, 1, 1, 1, 1, 1, 1, 1\big]^T,$$

we reach the minimizer of the optimization problem, i.e.

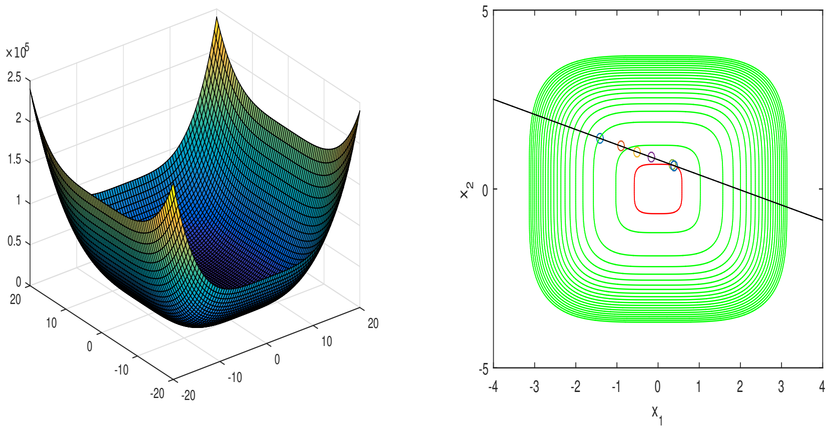
$$\big[-0.298, 1, 1, 1, -0.51, 0.004, -0.571, 0.004, -0.621, 0.005\big]^T,$$

with minimum value 1.1365 *in* 15 *iterations.*

Example 10. *We consider the following objective function in two dimensions*

$$\mathfrak{s}_1^4 + \frac{\mathfrak{s}_2^4}{2},$$

where $\phi_1 = \mathfrak{s}_1^4$ and $\phi_2 = \frac{\mathfrak{s}_2^4}{2}$. A single constraint is taken as $155\mathfrak{s}_1 + 367\mathfrak{s}_2 = 300$. Then executing the proposed Algorithm 3 with starting vector $[-5.963, -7.212]^T$, the minimum point $[-0.38896, 0.65316]^T$ having functional value 0.11389 is attained in 8 iterations. Figures 6a and 6b show the three-dimensional plot of the objective function and the progression of iteration points to the minimizer of the optimization problem side by side.



(a) 3D view **(b)** 2D view

Figure 6: Three dimensional plot of the objective function and progression of iteration points for Example 10

In both examples, the minimizers of the problems denote the optimal flow units and the objective functional value at the minimizer denotes the best flow satisfying the flow conservation.

5. CONCLUSION

A linear equality-constrained optimization problem ($\mathbb{ECOP}$) was considered, and an unconstrained reformulation (UR) using an exact penalty function was presented. To solve the reformulated problem, an alternative algorithm based on the sequential quadratic programming (SQP) framework was proposed. The modified SQP approach incorporated a positive definite Hessian modification, a regularized gap function, and a backtracking line search to improve robustness. The effectiveness of the proposed method was demonstrated through several numerical experiments. In addition, its practical applicability was illustrated by solving three real-world optimization problems.

As a future research direction, more rigorous numerical experiments and a wider range of real-world applications may be considered to further validate the performance and applicability of the proposed approach.

Funding: This research received no external funding.

REFERENCES

[1] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York: Springer, 2006.

[2] R. B. Wilson, "A simplicial algorithm for concave programming," *PhD Dissertation, Harvard University*, 1963.

[3] J. S. Alsumait, J. K. Sykulski, and A. K. Al-Othman, "A hybrid ga-ps-sqp method to solve power system valve-point economic dispatch problems," *Applied Energy*, vol. 87, no. 5, pp. 1773–1781, 2010.

[4] S. K. Chakraborty and G. Panda, "Two-phase-sqp method with higher-order convergence property," *Journal of the Operations Research Society of China*, vol. 4, no. 3, pp. 385–396, 2016.

[5] F. E. Curtis, T. Mitchell, and M. L. Overton, "A bfgs-sqp method for nonsmooth, nonconvex, constrained optimization and its evaluation using relative minimization profiles," *Optimization Methods and Software*, vol. 32, no. 1, pp. 148–181, 2017.

[6] P. E. Gill, V. Kungurtsev, and D. P. Robinson, "A stabilized sqp method: Global convergence," *IMA Journal of Numerical Analysis*, vol. 37, no. 1, pp. 407–443, 2017.

[7] A. F. Izmailov and M. V. Solodov, "Stabilized sqp revisited," *Mathematical Programming*, vol. 133, no. 1-2, pp. 93–120, 2012.

[8] P. T. Boggs and J. W. Tolle, "Sequential quadratic programming," *Acta Numerica*, vol. 4, pp. 1–51, 1995.

[9] N. I. M. Gould, M. E. Hribar, and J. Nocedal, "On the solution of equality constrained quadratic programming problems arising in optimization," *SIAM Journal on Scientific Computing*, vol. 23, no. 4, pp. 1376–1395, 2001.

[10] N. I. M. Gould and D. P. Robinson, "A second derivative sqp method: Global convergence," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 2023–2048, 2010.

[11] M. Fukushima, "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems," *Mathematical Programming*, vol. 53, no. 1-3, pp. 99–110, 1992.

[12] G. Auchmuty, "Variational principles for variational inequalities," *Numerical Functional Analysis and Optimization*, vol. 10, no. 9-10, pp. 863–874, 1989.

[13] A. Auslender, "Optimisation," *Méthodes numériques*, 1976.

[14] T. Larsson and M. Patriksson, "A class of gap functions for variational inequalities," *Mathematical Programming*, vol. 64, no. 1-3, pp. 53–79, 1994.

[15] M. Pappalardo, G. Mastroeni, and M. Passacantando, "Merit functions: a bridge between optimization and equilibria," *4OR*, vol. 12, no. 1, pp. 1–33, 2014.

[16] W. Li and J. Peng, "Exact penalty functions for constrained minimization problems via regularized gap function for variational inequalities," *Journal of Global Optimization*, vol. 37, no. 1, pp. 85–94, 2007.

[17] R. Sadhu and C. Nahak, "An exact convex reformulation of the quadratic programming," *Journal of Advanced Mathematical Studies*, vol. 12, no. 2, pp. 230–239, 2019.

[18] R. Sadhu, C. Nahak, and S. P. Dash, "Unconstrained reformulation of sequential quadratic programming and its application in convex optimization," *Proceedings of Indo-French Seminar*, 2020.

[19] P. T. Harker and J. S. Pang, "Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications," *Mathematical Programming*, vol. 48, no. 1-3, pp. 161–220, 1990.

[20] A. Beck, *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*. SIAM, 2014.

[21] J. R. Bunch, L. Kaufman, and B. N. Parlett, "Decomposition of a symmetric matrix," *Numerische Mathematik*, vol. 27, no. 1, pp. 95–109, 1976.

[22] I. S. Duff and J. K. Reid, "The multifrontal solution of indefinite sparse symmetric linear," *ACM Transactions on Mathematical Software*, vol. 9, no. 3, pp. 302–325, 1983.

[23] R. Fourer and S. Mehrotra, "Solving symmetric indefinite systems in an interior-point method for linear programming," *Mathematical Programming*, vol. 62, no. 1-3, pp. 15–39, 1993.

[24] F. E. Curtis, T. Mitchell, and M. L. Overton, "A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 474–500, 2012.

[25] M. Rashidi and M. Soleimani-Damaneh, “An exact penalty method with nonmonotone line search and rapid infeasibility detection for constrained multiobjective optimization: Application in supervised machine learning,” *Computers & Operations Research*, vol. 188, p. 107351, 2026.

[26] ——, “MultiSQP-GS: A sequential quadratic programming algorithm via gradient sampling for nonsmooth constrained multiobjective optimization,” *Computational Optimization and Applications*, vol. 89, no. 3, pp. 729–767, 2024.

[27] S. Surjanovic and D. Bingham, “Virtual library of simulation experiments: Test functions and datasets,” 2020, retrieved December 29, 2020 from http://www.sfu.ca/ ssurjano.

[28] Y. X. Yuan, “Subspace techniques for nonlinear optimization,” *Some Topics in Industrial and Applied Mathematics*, vol. 2007, pp. 206–218, 2007.

[29] J. H. Lee, Y. M. Jung, Y. Z. Yuan, and S. Yun, “A subspace sqp method for equality constrained optimization,” *Computational Optimization and Applications*, vol. 74, no. 1, pp. 177–194, 2019.

[30] E. D. Dolan and J. J. Moré, “Benchmarking optimization software with performance profiles,” *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.