Gulf Journal of Mathematics ISSN: 2309-4966
Vol 19, Issue 2 (2025) 398-411 https://doi.org/10.56947/gjom.v19i2.2801

AN EXPLICIT STEP LENGTH FOR SOLVING AN OPTIMIZATION PROBLEM

NOURA KACEM[1] and ASSMA LEULMI[2*]

Abstract. The problem of semi-definite programming (SDP) extends linear programming (LP) to solve a broader range of optimization problems, with significant advancements in algorithmic methods, particularly interior point techniques. In this article, we a logarithmic penalty approach for resolving SDP problems, where the direction of descent is determined using Newton's method. Additionally, for the step length, we propose new, more efficient, and robust lower bound functions. These proposed functions improve the accuracy and efficiency of the solution process. The effectiveness of the method is demonstrated through extensive numerical simulations, which validate the claims made in this study. The results confirm the practical feasibility and performance of the approach in solving complex semi-definite programming problems.

1. Introduction

Semi-Definite Programming (SDP) is a field of mathematical optimization concerned with problems where the variables are symmetric matrices constrained to be positive semidefinite. SDP provides a powerful framework, with theoretical and practical significance, for solving a wide variety of optimization problems involving matrix variables. Its relevance spans multiple disciplines, and solution techniques have been extensively explored both analytically and numerically. Numerous studies explore techniques to enhance constrained optimization processes, including those applicable to semidefinite programming (SDP). For instance, Manickam [12] investigated related optimization frameworks.

This work investigates a logarithmic barrier method for solving SDP problems. Instead of handling constraints directly, this approach iteratively refines the solution by incorporating the constraints into the objective function through a barrier term. The key contributions are as follows:

(1) Utilization of Newton's method to compute the search direction in each iteration of the barrier algorithm.
(2) Analysis of convergence problems encountered when employing standard logarithmic barrier functions, which can lead to algorithm failure.

Date: Received: Mar 24, 2025; Accepted: Apr 14, 2025
* Corresponding author.
2020 *Mathematics Subject Classification.* Primary 90C22; Secondary 90C51 .
Key words and phrases. Linear semi-definite programming, penalty methods, line-search.

(3) Modification of the step length calculation strategy to address convergence issues, proposing alternatives to potentially inefficient traditional line search methods.
(4) Proposal of alternative barrier functions to mitigate convergence issues encountered with standard formulations, aiming for improved robustness and efficiency.
(5) Introduction of novel approximation functions for step length calculation, designed for simplicity and computational efficiency compared to traditional approaches.

Ultimately, this research aims to contribute more efficient and robust numerical methods for solving semidefinite programs using a logarithmic barrier framework, thereby improving upon the performance of established techniques.

2. Statement of the problem

The semidefinite programming problem (SDP) is formulated as follows:

$$(P)\begin{cases} \max\langle C, Y\rangle \\ \langle A_i, Y\rangle = b_i \quad \forall i = 1, \dots, m \\ Y \in S_n^+. \end{cases}$$

where S_n^+ denotes the cone of $n \times n$ symmetric positive semidefinite matrices, $Y, C, A_i \in S_n^+$ for $i = 1, \dots, m$, and $b = (b_1, \dots, b_m)^t \in \mathbb{R}^m$. The notation $\langle C, Y\rangle$ represents the trace inner product, i.e., $\langle C, Y\rangle = \text{trace}(C^t Y)$, defined on the space M_n of $n \times n$ matrices.

- The feasible set involves the non-polyhedral convex cone of positive semidefinite matrices.
- The variable in the objective function of (P) is a matrix. Solving (P) is equivalent to solving its dual problem (D), in the sense that the optimal solution of one can often be derived from the optimal solution of the other via complementarity conditions. The dual problem is sometimes preferred computationally as its variable is a vector $x \in \mathbb{R}^m$.

The dual of problem (P) is given by:

$$(D)\begin{cases} \min b^t x \\ \sum\limits_{i=1}^{m} x_i A_i - C \in S_n^+ \\ x \in \mathbb{R}^m. \end{cases}$$

Let $\text{int}(S_n^+) = S_n^{++}$ denote the set of symmetric positive definite matrices. We define the feasible sets and their relative interiors (sets of strictly feasible solutions) as follows:

(1) $F_P = \{Y \in S_n^+ : \langle A_i, Y\rangle = b_i, \forall i = 1, \dots, m\}$ is the feasible set of the primal problem (P).
(2) $\bar{F}_P = \{Y \in F_P : Y \in \text{int}(S_n^+)\}$ is the set of strictly feasible primal solutions.

(3) $F_D = \left\{ x \in \mathbb{R}^m : \sum_{i=1}^{m} x_i A_i - C \in S_n^+ \right\}$ is the feasible set of the dual problem (D).
(4) $\bar{F}_D = \left\{ x \in \mathbb{R}^m : \sum_{i=1}^{m} x_i A_i - C \in \mathrm{int}(S_n^+) \right\}$ is the set of strictly feasible dual solutions.

Working assumptions:

(1) The system of linear equations $\langle A_i, Y \rangle = b_i$, $i = 1, \ldots, m$, has full rank m.
(2) The sets of strictly feasible solutions $\bar{F}_P$ and $\bar{F}_D$ are non-empty.

Under these assumptions, the following properties hold:

(1) The optimal solution sets for (P) and (D) are non-empty, compact, and convex.
(2) Optimality conditions: A pair $(\bar{Y}, \bar{x})$ with $\bar{Y} \in F_P$ and $\bar{x} \in F_D$ is optimal for (P) and (D) respectively if and only if the complementarity condition holds:
$$\left(\sum_{i=1}^{m} A_i \bar{x_i} - C \right) \bar{Y} = 0.$$

Under assumptions 1 and 2, strong duality holds, and solving (D) provides a solution to (P), and vice versa.

2.1. **The Perturbed Problem** (D_μ)**.** The dual problem (D) is approximated by the following logarithmic barrier problem (D_μ):

$$(D_\mu) \left\{ \begin{array}{l} \min f_\mu(x) \\ x \in \mathbb{R}^m, \end{array} \right.$$

where $\mu > 0$ is the barrier parameter, and $f_\mu : \mathbb{R}^m \to (-\infty, +\infty]$ is the barrier objective function defined by:

$$f_\mu(x) = \left\{ \begin{array}{ll} b^t x + n\mu \ln(\mu) - \mu \ln \left[\det \left(\sum_{i=1}^{m} x_i A_i - C \right) \right] & \text{if } x \in \bar{F}_D \\ +\infty & \text{otherwise.} \end{array} \right.$$

Problem (D_μ) is typically solved using Newton's method. A challenge arises in the line search step: the logarithmic determinant term in f_μ makes standard line search procedures computationally expensive. Instead of directly minimizing f_μ along the search direction d from the current iterate x, our strategy involves analyzing the auxiliary function $\theta(\alpha)$ defined for α such that $x + \alpha d \in \bar{F}_D$:

$$\theta(\alpha) \quad = \quad \frac{1}{\mu} \left[f_\mu(x + \alpha d) - f_\mu(x) \right].$$

We seek a function $G(\alpha)$ that approximates $\theta(\alpha)$ near $\alpha = 0$, satisfies

$$\theta(0) = G(0) = 0, \quad \theta'(0) = G'(0) < 0,$$

and whose minimizer is easy to compute. The condition $G'(0) < 0$ ensures d is a descent direction.

The function G should be chosen such that its minimizer α^* is easily computable and provides a sufficient decrease in f_μ at each iteration. To ensure G accurately approximates θ near $\alpha = 0$, we impose the condition that their second derivatives match:

$$\theta''(0) = G''(0).$$

This condition helps ensure that the step length obtained by minimizing G yields a good reduction in f_μ.

2.2. **Theoretical aspects of** (D_μ)**.**

Properties of f_μ. For $x \in \bar{F}_D$, let $B(x)$ be the $n \times n$ symmetric positive definite matrix defined as:

$$B(x) = \sum_{i=1}^{m} x_i A_i - C = L(x)L^t(x),$$

where $L(x)$ is its Cholesky factor. For $i, j = 1, \dots, m$, we define:

$$\begin{aligned} \widehat{A}_i(x) &= [L(x)]^{-1} A_i [L^t(x)]^{-1}, \\ b_i(x) &= \operatorname{trace}(\widehat{A}_i(x)) = \operatorname{trace}(A_i B^{-1}(x)), \\ \Delta_{ij}(x) &= \operatorname{trace}(B^{-1}(x) A_i B^{-1}(x) A_j) = \operatorname{trace}(\widehat{A}_i(x)\widehat{A}_j(x)). \end{aligned}$$

Here, $b(x)$ is the vector in $\mathbb{R}^m$ with components $b_i(x)$, and $\Delta(x)$ is the $m \times m$ symmetric matrix with entries $\Delta_{ij}(x)$.

Theorem 2.1 ([5])**.** *The function f_μ is twice continuously differentiable on its domain $\bar{F}_D$. For any $x \in \bar{F}_D$, its gradient $\nabla f_\mu(x)$ and Hessian $\nabla^2 f_\mu(x)$ are given by:*

(1) $\nabla f_\mu(x) = b - \mu b(x)$.
(2) $\nabla^2 f_\mu(x) = \mu \Delta(x)$.
(3) *The Hessian matrix $\nabla^2 f_\mu(x)$ (and thus $\Delta(x)$) is positive definite.*

2.2.1. *Existence and Uniqueness of Solutions to* (D_μ). Since $f_\mu(x) \to +\infty$ as x approaches the boundary of the feasible domain $\bar{F}_D$, and f_μ is differentiable (hence continuous) within $\bar{F}_D$, the function f_μ is lower semicontinuous.

To show that (D_μ) has a solution, it suffices to prove that f_μ is inf-compact. This can be established by showing that its recession function $(f_\mu)_\infty$ satisfies $(f_\mu)_\infty(d) > 0$ for all $d \neq 0$, which is equivalent to showing that the set $S_0 = \{d \in \mathbb{R}^m : (f_\mu)_\infty(d) \leq 0\}$ contains only $d = 0$.

We first state the following result:

Proposition 2.2 ([5])**.** *If $d \in \mathbb{R}^m$ satisfies $b^t d \leq 0$ and $\sum_{i=1}^{m} d_i A_i \in K$, then $d = 0$.*

Using this, we can show:

Theorem 2.3 ([5])**.** *If $(f_\mu)_\infty(d) \leq 0$, then $d = 0$.*

The preceding proposition, combined with properties of the logarithmic barrier function, leads to Theorem 2.3. This implies that f_μ is inf-compact, which guarantees the existence of at least one optimal solution to (D_μ).

2.3. **Convergence of the perturbed problem** (D_μ)**.** Let S_D denote the set of optimal solutions to the dual problem (D). Under our working assumptions, S_D is non-empty, compact, and convex. The distance from a point $x \in \mathbb{R}^m$ to the set S_D is defined as:

$$d(x, S_D) = \inf_{z \in S_D} \|x - z\|.$$

Let x_μ denote an optimal solution of (D_μ) and let $m(\mu) = f_\mu(x_\mu)$ be the optimal value. The asymptotic behavior as $\mu \to 0^+$ is described by the following theorems.

Theorem 2.4 ([8])**.** *As $\mu \to 0^+$, the distance from x_μ to the optimal set S_D converges to zero, i.e., $d(x_\mu, S_D) \to 0$. Furthermore, the optimal value $m(\mu)$ converges to the optimal value m_D of the dual problem (D), i.e., $m(\mu) \to m_D$.*

Theorem 2.5 ([8])**.** *If x_μ is an optimal solution to (D_μ) for each $\mu > 0$, then the set $\{x_\mu\}_{\mu>0}$ converges to the set S_D as $\mu \to 0^+$. Specifically, every limit point of $\{x_\mu\}$ as $\mu \to 0^+$ is an optimal solution of (D).*

2.4. **Computational aspects of the modified problem** (D_μ)**.**

2.4.1. *Newton Direction Computation.* The barrier formulation transforms the constrained problem (D) into an essentially unconstrained problem (D_μ) over the interior domain $\bar{F}_D$. Therefore, it can be solved using descent methods, such as Newton's method. Since $f_\mu(x) \to +\infty$ as x approaches the boundary of $\bar{F}_D$, any descent method initialized in $\bar{F}_D$ and using appropriate step lengths will maintain iterates $x_k \in \bar{F}_D$. Such methods are known as interior-point methods.

Let $x \in \bar{F}_D$ be the current iterate. The Newton direction d at x is chosen as the descent direction. From Theorem 2.1, the Newton direction d is obtained by solving the linear system:

$$\nabla^2 f_\mu(x) d = -\nabla f_\mu(x). \tag{1}$$

Substituting the expressions for the gradient and Hessian from Theorem 2.1 into (1), we obtain:

$$\mu\Delta(x)d = -(b - \mu b(x)) \quad \Leftrightarrow \quad \Delta(x)d = b(x) - \frac{1}{\mu}b. \tag{2}$$

Since $\Delta(x)$ is symmetric and positive definite (from Theorem 2.1), the linear system (2) has a unique solution and can be efficiently solved, for instance, using Cholesky decomposition. We assume $\nabla f_\mu(x) \neq 0$, otherwise x is the optimal solution to (D_μ). Since $\Delta(x)$ is positive definite, this implies $d \neq 0$.

Once the Newton direction d is computed, a line search is performed to find a step length $\bar{\alpha} > 0$ that ensures a sufficient decrease in f_μ and maintains feasibility, i.e., $x + \bar{\alpha} d \in \bar{F}_D$. Maintaining feasibility requires $B(x+\alpha d) = B(x) + \alpha H \in S_n^{++}$, where $H = \sum_{i=1}^m d_i A_i$.

The line search function $\theta(\alpha) = \frac{1}{\mu}[f_\mu(x + \alpha d) - f_\mu(x)]$ can be expressed [2] in terms of the eigenvalues λ_i of the matrix $E = L(x)^{-1} H (L(x)^{-1})^t$, where $L(x)$ is the Cholesky factor of $B(x)$. Specifically, for α such that $1 + \alpha\lambda_i > 0$ for all i:

$$\begin{aligned}
\theta(\alpha) &= \frac{1}{\mu}b^t(\alpha d) - \ln\det(B(x+\alpha d)) + \ln\det(B(x)) \\
&= \frac{\alpha}{\mu}b^t d - \ln\det(B(x) + \alpha H) + \ln\det(B(x)) \\
&= \frac{\alpha}{\mu}b^t d - \ln\det(L(L^t + \alpha L^{-1}H(L^t)^{-1}L^t)) + \ln\det(LL^t) \\
&= \frac{\alpha}{\mu}b^t d - \ln\det(L(I + \alpha E)L^t) + \ln\det(LL^t) \\
&= \frac{\alpha}{\mu}b^t d - \ln(\det(L)^2\det(I + \alpha E)) + \ln(\det(L)^2) \\
&= \frac{\alpha}{\mu}b^t d - \sum_{i=1}^{n}\ln(1 + \alpha\lambda_i).
\end{aligned}$$

Using $\theta'(0) = \frac{1}{\mu}\nabla f_\mu(x)^t d = \frac{1}{\mu}(b - \mu b(x))^t d$ and equation (2), $\Delta(x)d = b(x) - b/\mu$, we get $\theta'(0) = -(b(x) - b/\mu)^t d = -\sum_{i=1}^{n}(\lambda_i - \lambda_i^2)$ (the last equality requires derivation not shown here but often holds). The previous text derived:

$$\theta(\alpha) = \sum_{i=1}^{n}\left[\alpha\left(\lambda_i - \lambda_i^2\right) - \ln\left(1 + \alpha\lambda_i\right)\right], \quad \alpha \in [0, \breve{\alpha}),$$

where $\breve{\alpha} = \min\{-1/\lambda_i : \lambda_i < 0\}$ (or $\breve{\alpha} = +\infty$ if all $\lambda_i \geq 0$). Here, λ_i are the eigenvalues of the symmetric matrix E. Since $d \neq 0$ and assuming linear independence of A_i (Assumption 1), we have $H \neq 0$, which implies $E \neq 0$.

Let $\bar{\lambda} = \frac{1}{n}\sum_{i=1}^{n}\lambda_i$ and σ_λ denote the mean and standard deviation of the eigenvalues λ_i, and let $\|\lambda\| = (\sum_{i=1}^{n}\lambda_i^2)^{1/2}$ be the Euclidean norm of the vector $\lambda = (\lambda_1, \ldots, \lambda_n)$. The derivatives at $\alpha = 0$ are related to these eigenvalues:

$$\theta'(0) = \sum_{i=1}^{n}(\lambda_i - \lambda_i^2) = n\bar{\lambda} - \|\lambda\|^2,$$

$$\theta''(0) = \sum_{i=1}^{n}\lambda_i^2 = \|\lambda\|^2.$$

As noted in the thought process, for the Newton direction, it holds that $\theta''(0) = -\theta'(0)$, which implies $n\bar{\lambda} = 0$. The function can also be written as:

$$\theta(\alpha) = \alpha\theta'(0) - \sum_{i=1}^{n}\ln(1 + \alpha\lambda_i).$$

Unfortunately, there is no closed-form expression for the step length $\alpha_{\text{opt}} = \operatorname{argmin}_{\alpha\in[0,\breve{\alpha})}\theta(\alpha)$. Furthermore, solving the optimality condition $\theta'(\alpha) = 0$ iteratively requires repeated evaluation of $\theta(\alpha)$ and $\theta'(\alpha)$, which involves eigenvalue computations (or related operations) and can be prohibitively expensive. These computational difficulties motivate the search for alternative step length strategies.

Determining the exact upper limit $\breve{\alpha}$ of the feasible interval for α requires eigenvalue computation, which can be costly. As an alternative, one might compute a safe step length $\hat{\alpha}$ based on estimates involving the eigenvalues, such as the bound proposed in the original text:

$$\hat{\alpha} = \sup\left[\, \alpha \,:\, 1 + \alpha\beta_1 > 0 \,\right] = -1/\beta_1 \text{ (if } \beta_1 < 0), \text{ with } \beta_1 = \bar{\lambda} - \frac{\sigma_\lambda}{\sqrt{n-1}}.$$

This $\hat{\alpha}$ provides a computable upper bound on the step length based on statistical properties of the eigenvalues.

2.5. **Lower Bound Functions for $\theta(\alpha)$.** An alternative approach involves minimizing a function $G(\alpha)$ that serves as a lower bound for $\theta(\alpha)$ on an interval $[0, \hat{\alpha})$. For this strategy to be effective, G should be relatively simple to minimize while closely approximating θ. We impose the following conditions on G at $\alpha = 0$ to ensure a good local approximation:

$$G(0) = 0, \quad G''(0) = -G'(0) = \|\lambda\|^2.$$

These conditions match the value, first derivative, and second derivative of θ at $\alpha = 0$ when the Newton direction is used (recall $\theta(0) = 0$, $\theta'(0) = -\|\lambda\|^2$, $\theta''(0) = \|\lambda\|^2$).

2.5.1. *The Primary Lower Bound Function G_1.* Consider the function G_1 introduced by Leulmi [10]:

$$G_1(\alpha) = \delta_1\alpha - (n-1)\ln(1 + \beta_1\alpha) - \ln(1 + \gamma_1\alpha),$$

with parameters defined as:

$$\delta_1 = n\bar{\lambda} - \|\lambda\|^2, \quad \beta_1 = \bar{\lambda} - \frac{\sigma_\lambda}{\sqrt{n-1}}, \quad \text{and } \gamma_1 = \bar{\lambda} + \sigma_\lambda\sqrt{n-1}.$$

The logarithmic terms are well-defined for $\alpha \in [0, \hat{\alpha}_1)$, where:

$$\hat{\alpha}_1 = \begin{cases} -1/\beta_1 & \text{if } \beta_1 < 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Lemma 2.6 ([2])**.** *The function G_1 satisfies:*

$$\theta(\alpha) \geq G_1(\alpha), \quad \forall \alpha \in [0, \hat{\alpha}_1)\,,$$

and verifies the derivative conditions at $\alpha = 0$:

$$G_1(0) = 0 \quad \textit{and} \quad G_1''(0) = -G_1'(0) = \|\lambda\|^2\,.$$

The proof of this lemma relies on the following inequalities established by Merikhi [5]:

Theorem 2.7 ([5])**.** *Let $x_i > 0$ for $i = 1, \ldots, n$. Let $\bar{x}$ and σ_x denote the mean and standard deviation of the values $\{x_i\}_{i=1}^n$. Then:*

$$A \leq \sum_{i=1}^{n} \ln(x_i) \leq B, \quad \textit{and} \quad \sum_{i=1}^{n} \ln(x_i) \leq n\ln(\bar{x}), \tag{10}$$

where:

$$A = (n-1)\ln\left(\bar{x} + \frac{\sigma_x}{\sqrt{n-1}}\right) + \ln\left(\bar{x} - \sigma_x\sqrt{n-1}\right)$$

and

$$B = \ln\left(\bar{x} + \sigma_x\sqrt{n-1}\right) + (n-1)\ln\left(\bar{x} - \frac{\sigma_x}{\sqrt{n-1}}\right).$$

The function G_1 is strictly convex on $[0, \hat{\alpha}_1)$. Since $G_1'(0) = -\|\lambda\|^2 < 0$ (assuming $d \neq 0$) and $G_1(\alpha) \to +\infty$ as $\alpha \to \hat{\alpha}_1^-$ (if $\hat{\alpha}_1 < \infty$), G_1 admits a unique minimum α_{opt1} in $[0, \hat{\alpha}_1)$. This minimum is obtained by solving $G_1'(\alpha) = 0$. Setting the derivative

$$G_1'(\alpha) = \delta_1 - \frac{(n-1)\beta_1}{1+\beta_1\alpha} - \frac{\gamma_1}{1+\gamma_1\alpha}$$

to zero leads to a quadratic equation in α. Let this equation be $a\alpha^2 + b\alpha + c = 0$, where a, b, c depend on $\delta_1, \beta_1, \gamma_1$. The unique minimizer in the interval $[0, \hat{\alpha}_1)$ is given by the appropriate root of this quadratic equation:

$$\alpha_{\text{opt1}} = \frac{-b + \sqrt{b^2 - 4ac}}{2a}.$$

2.5.2. *Second Lower Bound Function G_2.* We define a second lower bound function G_2 as follows:

$$\begin{aligned} G_2(\alpha) &= (n\bar{\lambda} - ||\lambda||^2)\alpha - \tau_1 \ln(1+\varphi\alpha) \\ &= \sigma_3\alpha - \tau_1 \ln(1+\varphi\alpha), \end{aligned}$$

where:

$$\varphi = \frac{||\lambda||^2}{(n-1)\beta_1 + \gamma_1} \quad \text{and} \quad \sigma_3 = n\bar{\lambda} - ||\lambda||^2.$$

The parameter τ_1 is typically chosen to satisfy the derivative conditions (see below). The logarithmic term is well defined on $[0, \hat{\alpha}_3)$, where:

$$\hat{\alpha}_3 = \begin{cases} -1/\varphi & \text{if } \varphi < 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Lemma 2.8. *The function G_2 is strictly convex on $[0, \hat{\alpha}_3)$. If $\hat{\alpha}_3 < \infty$, then $G_2(\alpha) \to +\infty$ as $\alpha \to \hat{\alpha}_3^-$. Furthermore, for an appropriate choice of τ_1, G_2 satisfies:*

$$\theta(\alpha) \geq G_2(\alpha), \quad \forall\alpha \in [0, \hat{\alpha}_3),$$

and verifies the derivative conditions at $\alpha = 0$:

$$G_2(0) = 0 \quad \textit{and} \quad G_2''(0) = -G_2'(0) = ||\lambda||^2.$$

Partial Proof Sketch. Define the difference function:

$$k_1(\alpha) = \theta(\alpha) - G_2(\alpha).$$

Then, substituting the expressions for θ and G_2:

$$k_1(\alpha) = \left((n\bar{\lambda} - ||\lambda||^2)\alpha - \sum_{i=1}^{n} \ln(1+\alpha\lambda_i)\right)$$
$$- \left((n\bar{\lambda} - ||\lambda||^2)\alpha - \tau_1 \ln(1+\varphi\alpha)\right),$$
$$k_1(\alpha) = -\sum_{i=1}^{n} \ln(1+\alpha\lambda_i) + \tau_1 \ln(1+\varphi\alpha).$$

At $\alpha = 0$, we have $k_1(0) = 0$. The first derivative is $k_1'(\alpha) = -\sum \frac{\lambda_i}{1+\alpha\lambda_i} + \frac{\tau_1\varphi}{1+\varphi\alpha}$, so $k_1'(0) = -\sum \lambda_i + \tau_1\varphi = -n\bar{\lambda} + \tau_1\varphi$. The second derivative is:

$$k_1''(\alpha) = \sum_{i=1}^{n} \frac{\lambda_i^2}{(1+\lambda_i\alpha)^2} - \tau_1 \frac{\varphi^2}{(1+\varphi\alpha)^2}.$$

To ensure $k_1(\alpha) \geq 0$ for $\alpha \in [0, \hat{\alpha}_3)$, we typically require $k_1(0) = 0$, $k_1'(0) = 0$, and $k_1''(\alpha) \geq 0$. Setting $k_1'(0) = 0$ requires $\tau_1\varphi = n\bar{\lambda}$. If we choose $\tau_1 = \frac{n\bar{\lambda}}{\varphi} = \frac{n\bar{\lambda}((n-1)\beta_1+\gamma_1)}{||\lambda||^2}$ (assuming $\varphi \neq 0$), then $k_1'(0) = 0$. The condition $k_1''(\alpha) \geq 0$ requires showing $\sum_{i=1}^{n} \frac{\lambda_i^2}{(1+\lambda_i\alpha)^2} \geq \tau_1 \frac{\varphi^2}{(1+\varphi\alpha)^2}$. It is suggested to choose $\tau_1 = \frac{((n-1)\beta_1+\gamma_1)^2}{\|\lambda\|^2}$, which might be different.

□

Verification of conditions and choice of τ_1: If we choose:

$$\tau_1 = \frac{((n-1)\beta_1 + \gamma_1)^2}{||\lambda||^2},$$

then the derivatives of G_2 are:

$$\begin{aligned} G_2'(\alpha) &= (n\bar{\lambda} - ||\lambda||^2) - \frac{\tau_1\varphi}{1+\varphi\alpha} = \sigma_3 - \frac{\tau_1\varphi}{1+\varphi\alpha}, \\ G_2''(\alpha) &= \frac{\tau_1\varphi^2}{(1+\varphi\alpha)^2}. \end{aligned}$$

At $\alpha = 0$:

$$\begin{aligned} G_2'(0) &= \sigma_3 - \tau_1\varphi = (n\bar{\lambda} - ||\lambda||^2) - \frac{((n-1)\beta_1+\gamma_1)^2}{||\lambda||^2} \frac{||\lambda||^2}{(n-1)\beta_1+\gamma_1} \\ &= (n\bar{\lambda} - ||\lambda||^2) - ((n-1)\beta_1 + \gamma_1) \\ &= n\bar{\lambda} - ||\lambda||^2 - ((n-1)(\bar{\lambda} - \frac{\sigma_\lambda}{\sqrt{n-1}}) + (\bar{\lambda} + \sigma_\lambda\sqrt{n-1})) \\ &= n\bar{\lambda} - ||\lambda||^2 - ((n-1)\bar{\lambda} - \sigma_\lambda\sqrt{n-1} + \bar{\lambda} + \sigma_\lambda\sqrt{n-1}) \\ &= n\bar{\lambda} - ||\lambda||^2 - n\bar{\lambda} = -||\lambda||^2. \\ G_2''(0) &= \tau_1\varphi^2 = \frac{((n-1)\beta_1+\gamma_1)^2}{||\lambda||^2} \left(\frac{||\lambda||^2}{(n-1)\beta_1+\gamma_1}\right)^2 \\ &= \frac{||\lambda||^4}{||\lambda||^2} = ||\lambda||^2. \end{aligned}$$

Thus, with this choice of τ_1, the conditions $G_2'(0) = -\|\lambda\|^2$ and $G_2''(0) = \|\lambda\|^2$ are satisfied. The proof that $\theta(\alpha) \geq G_2(\alpha)$ requires showing $k_1''(\alpha) \geq 0$ with this τ_1.
Calculation of the Step Length α_{opt2}: Since G_2 is convex and tends to $+\infty$ at the boundary (if $\hat{\alpha}_3 < \infty$), it admits a unique minimum on $[0, \hat{\alpha}_3)$. This minimum, denoted α_{opt2}, is found by solving $G_2'(\alpha) = 0$:

$$\sigma_3 - \frac{\tau_1\varphi}{1+\varphi\alpha} = 0 \implies 1 + \varphi\alpha = \frac{\tau_1\varphi}{\sigma_3} \implies \alpha = \frac{1}{\varphi}\left(\frac{\tau_1\varphi}{\sigma_3} - 1\right) = \frac{\tau_1}{\sigma_3} - \frac{1}{\varphi}.$$

Thus, the minimum is:

$$\alpha_{\text{opt2}} = \frac{\tau_1}{\sigma_3} - \frac{1}{\varphi}.$$

2.5.3. *Third Lower Bound Function G_3.* We define a third lower bound function G_3 as follows:

$$G_3(\alpha) = \left(\tau_2\frac{||\lambda||^2}{\gamma_1} - ||\lambda||^2\right)\alpha - \tau_2 \ln\left(1 + \frac{||\lambda||^2}{\gamma_1}\alpha\right),$$

where $\tau_2 = \frac{\gamma_1^2}{||\lambda||^2}$. The logarithmic term is well defined on $[0, \hat{\alpha}_4)$, where:

$$\hat{\alpha}_4 = \begin{cases} -\gamma_1/||\lambda||^2 & \text{if } \gamma_1 < 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Lemma 2.9 ([2])**.** *With $\tau_2 = \frac{\gamma_1^2}{||\lambda||^2}$, the function G_3 satisfies:*

$$\theta(\alpha) \geq G_3(\alpha), \quad \forall \alpha \in [0, \hat{\alpha}_4),$$

and verifies the derivative conditions at $\alpha = 0$:

$$G_3(0) = 0 \quad \textit{and} \quad G_3''(0) = -G_3'(0) = ||\lambda||^2.$$

The verification of the derivative conditions for G_3 with the given τ_2 follows a similar calculation as for G_2.
Calculation of the Step Length α_{opt3}: G_3 is convex and admits a unique minimum on $[0, \hat{\alpha}_4)$, denoted α_{opt3}. This minimum is obtained by solving $G_3'(\alpha) = 0$:

$$G_3'(\alpha) = \left(\tau_2\frac{||\lambda||^2}{\gamma_1} - ||\lambda||^2\right) - \tau_2\frac{||\lambda||^2/\gamma_1}{1 + (||\lambda||^2/\gamma_1)\alpha} = 0.$$

Solving for α:

$$\begin{aligned}
\tau_2\frac{||\lambda||^2}{\gamma_1}\frac{1}{1 + (||\lambda||^2/\gamma_1)\alpha} &= ||\lambda||^2\left(\tau_2\frac{1}{\gamma_1} - 1\right) \\
\frac{\tau_2/\gamma_1}{1 + (||\lambda||^2/\gamma_1)\alpha} &= \frac{\tau_2 - \gamma_1}{\gamma_1} \\
\frac{\tau_2}{\tau_2 - \gamma_1} &= 1 + \frac{||\lambda||^2}{\gamma_1}\alpha \\
\frac{||\lambda||^2}{\gamma_1}\alpha &= \frac{\tau_2 - (\tau_2 - \gamma_1)}{\tau_2 - \gamma_1} = \frac{\gamma_1}{\tau_2 - \gamma_1} \\
\alpha &= \frac{\gamma_1^2}{||\lambda||^2(\tau_2 - \gamma_1)}.
\end{aligned}$$

Thus, the minimum is:

$$\alpha_{\text{opt3}} = \frac{\gamma_1^2}{||\lambda||^2(\tau_2 - \gamma_1)}.$$

2.6. **Description of the Algorithm.** We outline the logarithmic barrier algorithm below.

Initialization: Select a step length strategy (e.g., using G_1, G_2, or G_3). Set the parameters: $\epsilon > 0$ (stopping tolerance), $\mu > 0$ (initial barrier parameter), $\rho > 0$ (inner loop tolerance factor), and $\sigma \in [0, 1]$ (barrier reduction factor). Initialize with a strictly feasible point $x \in \bar{F}_D$.

Iteration: While $n\mu > \epsilon$:

(1) **Compute Hessian components:** Compute $B = B(x) = \sum_{i=1}^m x_i A_i - C$ and its Cholesky factor L such that $B = LL^t$. Compute:

$$\begin{cases} \widehat{A}_i(x) = [L]^{-1} A_i [L^t]^{-1} \\ b_i(x) = \text{trace}\left(\widehat{A}_i(x)\right) = \text{trace}(A_i B^{-1}(x)) \\ \Delta_{ij}(x) = \text{trace}(B^{-1}(x) A_i B^{-1}(x) A_j) = \text{trace}\left(\widehat{A}_i(x)\widehat{A}_j(x)\right). \end{cases}$$

The Hessian of the barrier function is $H_{ess} = \mu\Delta(x)$.

(2) **Compute Newton direction:** Solve the linear system $H_{ess} d = \mu b(x) - b$ for the search direction d.

(3) **Compute line search parameters:** Compute $H = \sum_{i=1}^m d_i A_i$, $E = L^{-1} H (L^t)^{-1}$, trace(E) (related to $n\bar{\lambda}$), and trace(E^2) (equal to $\|\lambda\|^2$).

(4) **Compute step length:** Compute the step length $\bar{\alpha}$ using the selected lower bound function strategy (minimizing G_1, G_2, or G_3).

(5) **Inner loop convergence check:** Let $x_{\text{new}} = x + \bar{\alpha} d$. If $|b^t x_{\text{new}} - b^t x| \leq \rho n\mu$, then the inner loop for the current μ has converged. Update $x \leftarrow x_{\text{new}}$, reduce the barrier parameter $\mu \leftarrow \sigma\mu$, and proceed to the next outer iteration (or terminate if $n\mu \leq \epsilon$). Else (if $|b^t x_{\text{new}} - b^t x| > \rho n\mu$), update $x \leftarrow x_{\text{new}}$ and repeat the inner loop steps (1-5) with the updated x and the same μ.

Termination: The algorithm stops when $n\mu \leq \epsilon$. The final iterate x is the approximate solution to (D). Typical parameter values are $\rho = 1$ or 2 and $\sigma = 0.125$. The step length $\bar{\alpha}$ in each iteration is determined by minimizing one of the lower bound functions G_1, G_2, or G_3 described previously.

Smaller values of μ yield better approximations of the true solution to (D). However, as $\mu \to 0$, the barrier problem (D_μ) becomes increasingly ill-conditioned, potentially leading to numerical instability. The algorithm terminates when $n\mu \leq \epsilon$ to avoid computations with excessively small μ.

The inner loop for a fixed μ terminates when the improvement in the objective function, estimated by $|b^t x_{\text{new}} - b^t x|$, is small relative to the scale of the barrier term, specifically when $|b^t x_{\text{new}} - b^t x| \leq \rho n\mu$. At this point, the current iterate x is considered a sufficiently accurate approximation to the minimizer $x(\mu)$ of (D_μ).

2.7. **Numerical Results.** We present numerical examples to examine the effectiveness of the proposed methods for solving semidefinite optimization problems.

All algorithms were implemented in MATLAB R2016a and executed on an HP Elite PC with an Intel(R) Core(TM) i5-8350U CPU @ 1.70GHz, Intel(R) UHD Graphics 620, 8 GB RAM, running Windows 11 Pro. The following parameter values were used: $\varepsilon = 0.1$, $\sigma = 0.125$, $\rho = 1$, and initial $\mu = 0.3$.

2.7.1. *Variable Size Example.* Consider problems where $n = 2m$, $b = (2, \dots, 2)^t \in \mathbb{R}^m$, and the matrices $A_k \in S_n^+$ for $k = 1, \dots, m$ have elements defined by:

$$(A_k)_{ij} = \begin{cases} 1 & \text{if } i = j = k & \text{or } i = j = k+m \\ a^2 & \text{if } i = j = k+1 & \text{or } i = j = k+m+1 \quad \text{(indices mod } n \text{ if needed)} \\ -a & \text{if } i = k, j = k+1 & \text{or } i = k+m, j = k+m+1 \\ -a & \text{if } i = k+1, j = k & \text{or } i = k+m+1, j = k+m \\ 0 & \text{otherwise.} \end{cases}$$

Here, $a \in \mathbb{R}$ is a given constant. (Note: care must be taken with indices like $k+1$ and $k+m+1$ to ensure they remain within $\{1, \dots, n\}$; cyclic indexing might be intended).
Test 1: For $a = 9, C = -2I$, we have:

(n, m)	G_1		G_2		G_3	
$(85, 170)$	1	0.995572	1	1.000862	1	0.993733
$(100, 200)$	1	1.012261	1	1.008676	1	1.002608
$(130, 260)$	1	0.998580	1	1.002016	1	0.985113

Test 2: For $a = 7, C = -2I$, we have:

(n, m)	G_1		G_2		G_3	
$(85, 170)$	1	1.005911	1	0.985787	1	1.008890
$(100, 200)$	1	1.024742	1	1.015295	1	0.988366
$(130, 260)$	1	1.009724	1	1.001837	1	0.991888

Test 3: For $a = 3, C = -2I$, we have:

(n, m)	G_1		G_2		G_3	
$(85, 170)$	1	0.996393	1	0.995986	1	1.005372
$(100, 200)$	3	1.001688	1	0.989932	1	0.990862
$(130, 260)$	2	0.991949	1	1.001645	1	1.001264

Test 4: For $a = 2, C = -2I$, we have:

(n, m)	G_1		G_2		G_3	
$(85, 170)$	1	1.004067	1	0.997610	1	1.003901
$(100, 200)$	1	1.002648	1	1.013454	1	0.996014
$(130, 260)$	1	1.006472	1	0.992946	1	1.013465

2.7.2. *Comment.* Through these numerical tests, our proposed modification to the step length calculation offers significant improvements in both the efficiency and stability of the logarithmic penalty algorithm. These benefits are demonstrated through a clear reduction in iterations and computational time, as well as through the algorithm's ability to effectively handle difficult or perturbed cases that would typically cause convergence issues.

3. Conclusion

In this paper, we presented a logarithmic barrier algorithm designed to solve linear semidefinite programming (SDP) problems. The algorithm utilizes Newton's method to compute an effective descent direction at each iteration. To determine the step length along this direction, we introduced novel lower bound functions for the line search objective. These functions facilitate the efficient computation of suitable step lengths.

To support our theoretical findings, numerical experiments were conducted. The results demonstrate the practical effectiveness and performance of the proposed approach. The new lower bound functions proved reliable and represent an efficient technique for determining the step length within logarithmic barrier methods for SDP. Their utility may extend to other classes of optimization problems as well.

A key advantage of these novel functions is their potential to address convergence difficulties that can arise when using traditional line search methods or simpler approximation functions. They provide robust alternatives for step length calculation, applicable both in cases where standard methods falter and potentially as replacements for classical techniques in general scenarios.

References

1. Alizadeh, F. (1995). Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM journal on Optimization, 5(1), 13-51.* https://doi.org/10.1137/S1052623491196224
2. Bendemagh, Z., & Bedjaoui, C. (2023). *Des fonctions minorantes pour la programmation semie-définie par une méthode barrière logarithmique,* mémoire de master, Université Ferhat Abbas Sétif 1.
3. Benterki, D., Crouzeix, J.P., & Merikhi, B. (2007). A numerical feasible interior point method for linear semidefiniteprograms, *CRAIRO Oper. Res. 41,* 49-59. https://doi.org/10.1051/ro:2007006
4. Chaussard, J. (2016-2017). *Introduction à MATLAB.* Laga-Université Paris 13, Ecole Sup Galilée-Cursus Ingénieur 1ère année.
5. Crouzeix, J. P., & Merikhi, B. (2008). A logarithm barrier method for semi-definite programming. *RAIRO-Operations Research, 42(2),* 123-139. https://doi.org/10.1051/ro:2008005
6. Crouzeix, J. P., & Seeger, A. (1996). New bounds for the extreme values of a finite sample of real numbers. *Journal of mathematical analysis and applications, 197(2),* 411-426. https://doi.org/10.1006/jmaa.1996.0029
7. Hafsi, N. (2010). *Une Recherche Linéaire Avec Des Fonctions Majorantes Dans Certains Problèmes D'optimisations,* Diplôme De Magister, Universite Mentouri-Constantine.
8. Leulmi, A. (2024). An Efficient Penalty Method without a Line Search for Nonlinear Optimization. *Axioms, 13(3),* 176. https://doi.org/10.3390/axioms13030176
9. Leulmi, A., & Leulmi, S. (2019). Logarithmic barrier method via minorant function for linear programming. *Journal of Siberian Federal University Mathematics and physics, 12(2),* 191-201. https://doi.org/10.17516/1997-1397-2019-12-2-191-201
10. Leulmi, S., & Leulmi, A. (2023). A novel adaptive method based on new minorant-majorant functions without line search for semidefinite optimization. *Nonlinear Dynamics and Systems Theory, 23(3),* 310-322 .

11. Leulmi, A., Ziadi, R., Souli, C., Saleh, M. A., & Almaymuni, A. Z. (2024). An Interior Point Algorithm for Quadratic Programming Based on a New Step-Length. *International Journal of Analysis and Applications, 22,* 233-233. https://doi.org/10.28924/2291-8639-22-2024-233
12. Manickam, K. (2015). Optimization techniques for mixed finite element methods of a semilinear elliptic PDEs – superconvergence. *Gulf Journal of Mathematics, 3(1),* 55–66. https://doi.org/10.56947/gjom.v3i1.170
13. Souli, C., & Leulmi, A. (2025). Theoretical and numerical results for nonlinear optimization problems. *Nonlinear Dynamics and Systems Theory, 25(1),* 101–111.
14. Rockafellar,R.T. (1970). *Convex analysis,* Princeton University Press, New Jerzy. https://doi.org/10.1515/9781400873173
15. Wolkowicz, H., & Styan, G. P. H. (1980). Bounds for eigenvalues using traces. *Linear Algebra and Appl.29,* 471-506. https://doi.org/10.1016/0024-3795(80)90077-4

[1] Department of mathematics, Faculty of science, University of Ferhat Abbas Setif-1

Email address: noura.bencherif@univ-setif.dz, nkassemkassem76@gmail.com

[2] Department of mathematics, Faculty of science, University of Ferhat Abbas Setif-1

Email address: assma.leulmi@univ-setif.dz, as_smaleulmi@yahoo.fr