Numerical Optimization Assignment Report 2024/25

Equality Constrained Quadratic Programming

Riccardo Kiefer (ID: 301286)
Ilaria Palumbo (ID: 302642)

May 14, 2026

Contents

1 Introduction

This report focuses on solving an **Equality Constrained Quadratic Programming (QP)** problem using various numerical techniques. The given problem is formulated as:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} x^T Q x + c^T x \tag{1}$$

subject to the linear equality constraints:

$$Ax = b, \tag{2}$$

where: - $Q \in \mathbb{R}^{n \times n}$ is a symmetric positive semi-definite matrix. - $c \in \mathbb{R}^n$ is a linear term. - $A \in \mathbb{R}^{m \times n}$ is the constraint matrix. - $b \in \mathbb{R}^m$ represents the right-hand side of the constraints.

1.1 Karush-Kuhn-Tucker (KKT) Conditions

To solve this constrained optimization problem, we define the Lagrangian function:

$$\mathcal{L}(x, \lambda) = \frac{1}{2} x^T Q x + c^T x + \lambda^T (Ax - b), \tag{3}$$

where $\lambda \in \mathbb{R}^m$ is the vector of Lagrange multipliers.

The KKT conditions for optimality are derived by taking the first-order conditions:

$$\nabla_x \mathcal{L} = Qx + c + A^T \lambda = 0, \tag{4}$$

$$\nabla_\lambda \mathcal{L} = Ax - b = 0. \tag{5}$$

This results in the KKT linear system:

$$\begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}. \tag{6}$$

$$Qx + c + A^T \lambda = 0, \quad Ax = b \tag{7}$$

where: - Q is the Hessian matrix of the quadratic function. - c is the linear term. - A represents the equality constraints. - λ is the vector of Lagrange multipliers.

1.2 Problem-Specific Constraints

In this assignment, the quadratic programming problem is given by:

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n-1} x_i x_{i+1} + \sum_{i=1}^{n} x_i \tag{8}$$

subject to the set of equality constraints:

$$\sum_{j=0}^{M-1} x_{i+jK} = \varepsilon, \quad \text{for } i = 1, \ldots, K, \tag{9}$$

where K is the number of equality constraints, $M = n/K$ and ε is a randomly generated value in $(0, 1)$ with a seed set to the minimum student ID.

1.3 Solution Strategies

This report explores multiple methods to solve the system (6):

- **Direct KKT Solution:** The system is solved using direct solvers. In general, when solving a linear system in MATLAB with a direct method, the `mldivide` function (equivalent to the operator \) is used.
- **Iterative Methods (GMRES):** The system is solved using iterative solvers, both with and without preconditioning. Since the system matrix KKT is not *positive definite*, gradient-based and conjugate gradient (CG) methods cannot be applied. Thus, the more general GMRES (Generalized Minimal Residual) method is chosen.
- **Schur Complement Method:** The system is solved both with and without explicitly computing Q^{-1}. This method requires Q to be invertible. First, the Schur complement matrix is computed as $S = AQ^{-1}A^T$. Then, $\lambda^\star$ is obtained by solving the linear system:

$$(AQ^{-1}A^T)\lambda = -b - AQ^{-1}c. \tag{10}$$

  Finally, the optimal solution $x^\star$ is computed by solving:

$$Qx^\star = -c - A^T\lambda^\star. \tag{11}$$

- **Null Space Method:** Assuming that A has full rank (i.e., $\text{rank}(A) = m$), let $Z \in \mathbb{R}^{n\times(n-m)}$ be a matrix whose columns form a basis for $\ker(A)$, and let $\widehat{x}$ be a particular solution of the equality constraint, i.e., $A\widehat{x} = b$. The null space method consists of making the substitution:

$$x = Zv + \widehat{x}. \tag{12}$$

  Then, the optimal $v^\star$ is obtained by solving:

$$(Z^TQZ)v = -Z^T(Q\widehat{x} + c). \tag{13}$$

  The optimal solution $x^\star$ is then computed as:

$$x^\star = Zv^\star + \widehat{x}. \tag{14}$$

  Finally, the Lagrange multiplier $\lambda^\star$ is obtained by solving:

$$(AA^T)\lambda = -A(c + Qx^\star). \tag{15}$$

The numerical results will analyze the efficiency, accuracy, and computational performance of these methods across different problem sizes.
In the last two methods, all linear systems were solved using MATLAB's direct solver. While iterative methods, both with and without preconditioning, could have been explored, the direct solver provided excellent performance across all cases. Given these strong results, further investigation into alternative solution methods was considered beyond the scope of this assignment.

2 Numerical Results and Discussion

2.1 Test Parameters

Given the sparsity of the matrices Q and A , they are stored using the `sparse` command in MATLAB. This significantly enhances memory efficiency and computational performance across the different algorithms. The table below illustrates how the condition number of the system matrix varies with the problem's dimensionality.

$(n,\ K)$	Condition Number of KKT Matrix
(2, 1)	6
(2000, 100)	2586501.5381
(20000, 500)	516873569.2777
(200000, 1000)	258421826526.4359

2.2 Performance Analysis

We compare the methods based on computational time, successful convergence to the optimal point satisfying the stationary conditions of the Lagrangian function, and the accuracy of the solution.

2.2.1 Precision of the solution

The tables below report the precision of the solution measured with respect to the norm of the Lagrangian gradient: the closer to zero the more precise.

- Case $n = 2, K = 1$:

Method	$\|\|x^\star\|\|_2$	$\|\|\nabla_x \mathcal{L}\|\|_2$	$\|\|\|\partial_\lambda \mathcal{L}\|\|_2$	Success
Direct	0.0154948173	2.2204e-16	1.1102e-16	yes
GMRES without preconditioning	0.0154948173	1.9860e-15	3.4694e-17	yes
GMRES with preconditioning	0.0154948173	2.2204e-16	2.2204e-16	yes
Schur Complement with Q^{-1}	0.0154948173	0	0	yes
Schur Complement without Q^{-1}	0.0154948173	0	0	yes
Null Space	0.0154948173	0	0	yes

- Case $n = 2000, K = 100$:

Method	$\|\|x^\star\|\|_2$	$\|\|\nabla_x \mathcal{L}\|\|_2$	$\|\|\nabla_\lambda \mathcal{L}\|\|_2$	Success
Direct	0.0536838738	4.1093e-15	4.1489e-17	yes
GMRES without preconditioning	0.0491750769	4.1148e-05	4.9794e-06	yes
GMRES with preconditioning	0.0536838716	1.0595e-11	4.2658e-05	yes
Schur Complement with Q^{-1}	0.0536838723	0	2.2263e-08	yes
Schur Complement without Q^{-1}	0.0536838792	0	2.5359e-08	yes
Null Space	0.0536838738	0.0049	1.6273e-17	yes

- Case $n = 20000, K = 500$:

Method	$\|\|x^\star\|\|_2$	$\|\|\nabla_x \mathcal{L}\|\|_2$	$\|\|\nabla_\lambda \mathcal{L}\|\|_2$	Success
Direct	0.0848752767	1.3816e-14	1.3211e-16	yes
GMRES without preconditioning	0.0775180676	1.2742e-04	1.4610e-05	yes
GMRES with preconditioning	0.0848749509	2.7561e-08	2.7561e-08	yes
Schur Complement with Q^{-1}	0.0848765884	0	4.6290e-04	yes
Schur Complement without Q^{-1}	0.0847954073	0	4.6290e-04	yes
Null Space	0.0848752767	0.0035	3.2546e-17	yes

- Case $n = 200000, K = 1000$:

Method	$\|\|x^\star\|\|_2$	$\|\|\nabla_x \mathcal{L}\|\|_2$	$\|\|\nabla_\lambda \mathcal{L}\|\|_2$	Success
Direct	0.0536757071	4.3994e-14	4.0935e-16	yes
GMRES without preconditioning	0.0490139663	3.9316e-04	2.6224e-05	yes
GMRES with preconditioning	0.0533633264	3.0866e-08	3.0866e-08	yes
Schur Complement without Q^{-1}	0.0488147158	0	1.3229	no
Null Space	0.0536756305	0.0015	3.7044e-17	yes

2.2.2 Computational time

The tables below present the time required for each method to compute the optimal solution, along with the number of iterations for iterative methods.

- Direct method:

n	K	Objective value	Iterations	Time	Success
2	1	0.0220330254	nan	0.014482 s	yes
2000	100	2.19129807560	nan	0.021744 s	yes
20000	500	10.9564903600	nan	0.078493 s	yes
200000	1000	21.9129807199	nan	1.231428 s	yes

- GMRES **without** preconditioning:

n	K	Objective value	Iterations	Time	Success
2	1	0.0220330254	2	0.005231 s	yes
2000	100	2.1912982122	61	0.018020 s	yes
20000	500	10.9564907740	37	0.125439 s	yes
200000	1000	21.9129815165	16	0.134643 s	yes

- GMRES **with** preconditioning:

n	K	Objective value	Iterations	Time	Success
2	1	0.0220330254	1	0.014482 s	yes
2000	100	2.1912980754	1	0.027017 s	yes
20000	500	10.9564903916	1	0.217174 s	yes
200000	1000	21.9129929008	5	40.195829 s	yes

- Schur Complement Method knowing Q^{-1}:

n	K	Objective value	Iterations	Time	Success
2	1	0.0220330254	nan	0.012395 s	yes
2000	100	2.1912980213	nan	0.076866 s	yes
20000	500	10.9566522818	nan	13.228560 s	yes
200000	1000	*Memory Fault*	nan	nan	no

- Schur Complement Method **without** knowing Q^{-1}:

n	K	Objective value	Iterations	Time	Success
2	1	0.0220330254	nan	0.008493 s	yes
2000	100	2.1912983110	nan	0.017413 s	yes
20000	500	10.9461435507	nan	0.172694 s	yes
200000	1000	-19.9210538681	nan	3.669513 s	no

- Null Space Method:

n	K	Objective value	Iterations	Time	Success
2	1	0.0220330254	nan	0.012679 s	yes
2000	100	2.1912980756	nan	0.020787 s	yes
20000	500	10.9564903600	nan	0.206291 s	yes
200000	1000	21.9129807199	nan	9.585635 s	yes

2.3 Visualization for $n = 2, K = 1$

For the case $n = 2, K = 1$, we visualize:

- The contour plot of the quadratic function, with the constraint and the optimal solutions. At this scale the difference in the solutions is not appreciable. All the optimal points seems to lie on the constraint line.

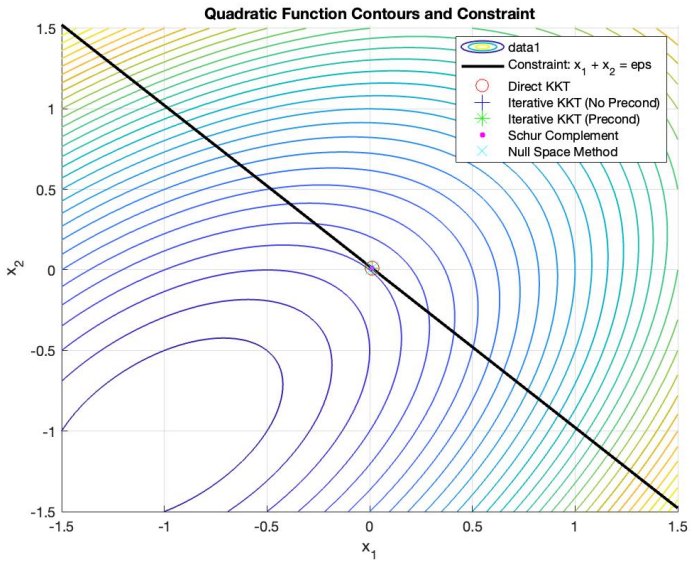

Figure 1: Contour plot, constraint line, and optimal solutions.

- The differences in the solutions starts to be notable at a scale of 10^{-15} as shown in the picture below. This scale is just one order of magnitude greater that the machine precision and thus the differences are negligible.

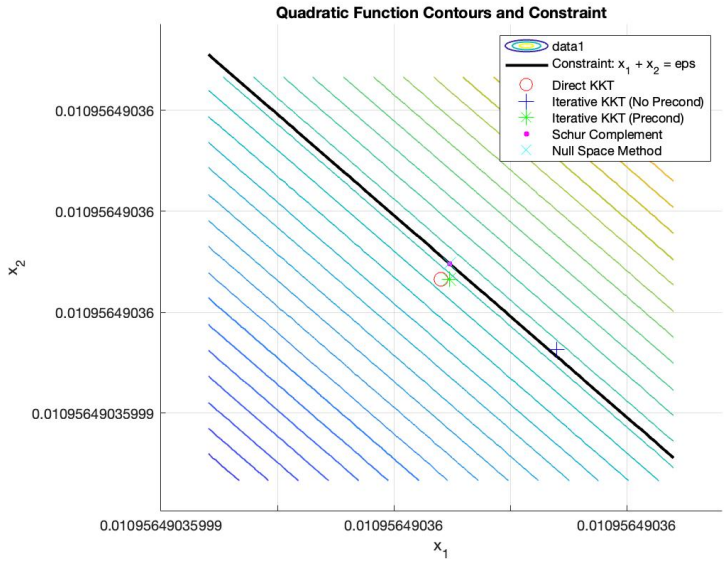

Figure 2: Enlargement around optimal solutions.

- The components of the optimal solutions obtained with the different methods:

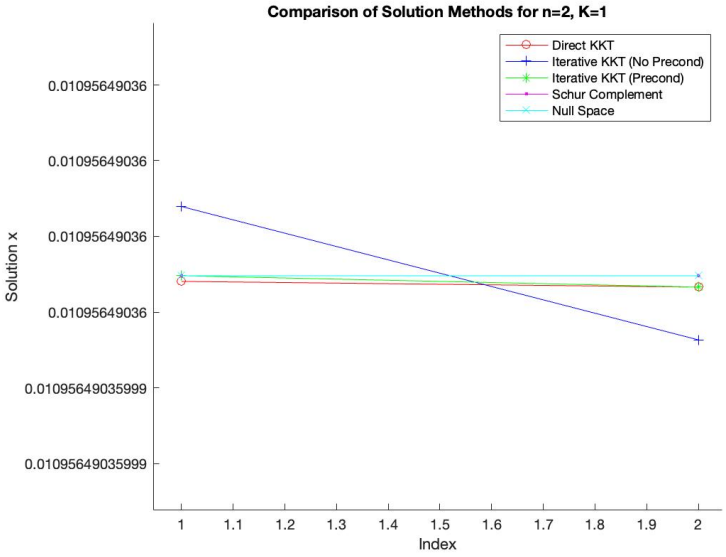

Figure 3: Components of optimal solutions.

3 Conclusions

This study examined different numerical methods to solve equality-constrained quadratic programming problems. The results indicate that:

- The direct method for solving the KKT linear system performs exceptionally well across all tested dimensions, consistently producing highly accurate solutions in a very short time. The success rate is 100 %.

- The iterative solvers also demonstrate excellent performance, achieving a 100% success rate. Preconditioning effectively reduces the number of iterations required for convergence and does improve the accuracy of the solution. However, the computational overhead required for the preconditioning (incomplete LU decomposition takes approximately 380 seconds for $n = 200000$) is not justified by a substantial increase in solution precision for this problem. Cholesky decomposition is not feasible since the system matrix is not positive definitive.

- The Schur complement method yields the poorest results. Explicitly computing Q^{-1} using `inv(Q)` takes more than 10 seconds for $n = 20000$ and is infeasible for $n = 200000$ due to memory limitations. Using the method without explicitly computing Q^{-1} is significantly faster for $n = 20000$ but fails to converge for $n = 200000$.

- The null space method also achieves a 100% success rate and outputs the most precise solutions, but it is computationally slower than both the direct method and GMRES without preconditioning, particularly for high-dimensional problems.

Considering the overall performance of all solvers, the direct method emerges as the best approach for this problem, providing an optimal balance between speed and simplicity. Iterative methods, both with and without preconditioning, perform significantly worse than the direct method in terms of both accuracy and computational efficiency. The null space method is a viable alternative; however, it requires additional theoretical and programming expertise, as it involves constructing the null space matrix and determining a particular solution manually