Notes on Optimization and Interior-Point Methods

Joe Koszut

University of California, Berkeley

December 2023

Abstract

The field of optimization has a wide span of applications ranging from image/signal processing and portfolio optimization to design engineering and control of dynamical systems. The area of convex optimization deals with problems where local optimality implies global optimality. Well-studied algorithms exist which can solve convex optimization problems reliably and quickly. The predictability of convex optimization techniques make them an enabling factor for using optimization in many practical applications. While the traits of convex optimization are broadly attractive, the specific problems which can be solved take on a variety of forms. The number of problem variables, equality constraints, and inequality constraints can range from single digits to hundreds of thousands, if not more. The variability in problem structure makes it difficult to design algorithms which work well for all problem types. The more general we make an algorithm, the more we tend to leave on the table in terms of solve time, and vice versa. In some situations, solve time is not a limiting factor in which case there is little to no benefit in using a faster algorithm. In other situations, solving an optimization problem within a small window of time can be an enabling factor for implementing a novel and performant idea. Interior-point methods (IPMs) have been a popular technique for solving convex optimization problems since the 1990s as many convex optimization problems can be posed as linear cone programs for which IPMs can be applied. The goal of these notes is to provide background into the field of optimization leading up to understanding the ideas behind interior-point methods and being able to think how certain problem structures can be exploited to design performant optimization algorithms.

Table of contents

Chapter 1

Introduction

1.1 History of Interior-Point Methods

Interior-point methods (IPMs) solve convex optimization problems by starting out with a strictly feasible point and moving towards the boundary of the feasible domain. This is opposed to, for example, the simplex method which traverses the vertices of the feasible region or a barrier penalty method which allows for iterates to go outside the boundary of the feasible domain and applies a penalty the further the iterates move away from the boundary.

The first IPM, known as *affine scaling*, was developed by Dikin in 1967 to solve linear programs (LPs). In 1979, Khachayin developed the first polynomial-time method, known as the *ellipsoid method*, for linear programming. It enabled finding guaranteed solutions to feasible problems which before had been only sometimes solvable. This was a major breakthrough and earned a spot on the front page of the New York Times with the heading, "A Soviet Discovery Rocks World of Mathematics". In 1984, Karmarkar developed an IPM which also had polynomial time complexity but was faster than Khachayin's method, both in worst-case complexity as well as in practice. This too earned a feature on the front page of the New York Times titled, "Breakthrough in Problem Solving". Dikin's method went largely unnoticed until it was realized that Karmarkar's method had similarities to Dikin's. The years following 1984 saw rapid development and expansion of interior-point methods.

A class of interior-point methods known as *primal-dual methods* came out on top, performing better than other interior-point methods on most practical problems. In 1989, Mehrotra developed a predictor-corrector method for primal-dual interior point methods which instead of solving for one search direction, it solved for two different directions: a predictor and a corrector. The final search direction ends up being somewhere between the predictor and corrector directions. Although Mehrotra's predictor-corrector method adds computational overhead, solving for two directions instead of one, it usually pays off with a reduction in the number of iterations required to achieve convergence. As a result, the

primal-dual Mehrotra predictor-corrector method is the basis for most interior-point algorithms in use today.

1.2 Outline

These notes are based on various sources although the majority of ideas come from the textbooks by Martins & Ning [1] and Wright [2]. The notes are organized as follows. In Chapter 2, unconstrained, equality-constrained, inequality-constrained, and equality-plus inequality-constrained optimization problems (which are not necessarily convex) are introduced. The ideas of KKT conditions and duality are also introduced. In Chapter 3, the field of linear programming is described. In Chapter 4, interior-point methods are introduced along with the idea of Mehrotra's predictor-corrector and homogeneous self-dual embedding. In Chapter 5, extensions are given of ideas from Chapter 4 to conic programming and further reading is discussed.

Chapter 2

General Concepts

2.1 Unconstrained Optimization

An unconstrained optimization problem can be written in the following form:

$$\min_{x} \qquad f(x)$$

where $f(x)$ is the objective function and $x \in \mathbb{R}^n$ is the n-dimensional vector of decision variables. If a minimum x^* of the objective function exists, it is a stationary point, i.e., $\nabla f(x^*) = 0$. The first-order Taylor series expansion of the objective function can be written as

$$f(x+p) \approx f(x) + \nabla f(x)^\top p \tag{2.2}$$

where p is small enough such that higher order terms are negligible. If x^* is a minimum point, then the objective at every point in a small neighborhood of x^*, denoted $\mathcal{N}_{x^*}$, must have a greater value, i.e.,

$$f(x^*+p) \geq f(x^*) \quad \forall p \in \mathcal{N}_{x^*} \tag{2.3}$$

Combining (2.2) and (2.3) requires that a minimum point satisfies

$$f(x^*) + \nabla f(x^*)^\top p \geq f(x^*) \implies \nabla f(x^*)^\top p \geq 0 \tag{2.4}$$

Therefore, in order for a point x^* to be a local minimum, i.e., any perturbation to x^* only increases the objective value, it is required to have $\nabla f(x^*)^\top p \geq 0$ for all $p \in \mathcal{N}_{x^*}$. Some comments can be made regarding the value of the term $f(x)^\top p$:

- $\nabla f(x)^\top p = 0$ defines a hyperplane that contains the directions along which the first-order variation of f is zero
- $\nabla f(x)^\top p < 0$ defines an open half-space of directions where f decreases
- $\nabla f(x)^\top p > 0$ defines an open half-space of directions where f increases

Theorem 1. *The minimum point for an unconstrained problem must satisfy*

$$\nabla f(x^*) = 0 \tag{2.5}$$

Proof. For an unconstrained problem, if there exists a direction p such that $\nabla f(x)^\top p > 0$, this means that there also exists a direction $-p$ such that $\nabla f(x)^\top (-p) < 0$. However, this violates (2.4). Therefore, we must have $\nabla f(x^*) = 0$. □

Definition. (2.5) *is known as the* ***first-order necessary optimality condition for an unconstrained optimization problem****.*

2.2 Equality-Constrained Optimization

An equality-constrained optimization problem can be written in the following form:

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & h(x) = 0 \end{aligned}$$

where $h(x) \in \mathbb{R}^{n_h}$. To achieve optimality, (2.4) still applies, but p must also be a feasible direction. The first-order Taylor series expansion for each equality constraint can be written as:

$$h_j(x+p) \approx h_j(x) + \nabla h_j(x)^\top p, \quad j = 1, \ldots, n_h \tag{2.7}$$

where p once again is small enough such that higher order terms are negligible. If x is a feasible point, then $h_j(x) = 0$, and the first-order Taylor series expansion from (2.7) reduces to

$$h_j(x+p) \approx \nabla h_j(x)^\top p, \quad j = 1, \ldots, n_h \tag{2.8}$$

Therefore, first-order feasibility requires that

$$\nabla h_j(x)^\top p = 0, \quad j = 1, \ldots, n_h \tag{2.9}$$

The interpretation behind (2.9) is that a direction is feasible when it is orthogonal to all equality constraint gradients. This can be written in matrix form as

$$J_h(x)p = 0 \tag{2.10}$$

where J_h is the null space of the Jacobian of the equality constraints.

Corollary 1. (2.10) *implies that any feasible direction p must lie in the null space of J_h, i.e., $p \in \mathcal{N}(J_h)$*[1]*.*

The matrix $J_h(x)$ can be represented as such

$$J_h(x) = \begin{bmatrix} \nabla h_1(x)^\top \\ \vdots \\ \nabla h_{n_h}(x)^\top \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \cdots & \frac{\partial h_1}{\partial x_{n_x}} \\ \vdots & & & \\ \frac{\partial h_{n_h}}{\partial x_1} & \frac{\partial h_{n_h}}{\partial x_2} & \cdots & \frac{\partial h_{n_h}}{\partial x_{n_x}} \end{bmatrix} \in \mathbb{R}^{n_h \times n_x} \tag{2.11}$$

Assuming that $J_h(x)$ has full row rank, i.e., the equality constraint gradients are linearly independent, it can be seen from (2.11) that $\mathcal{N}(J_h)$ has dimension $n_x - n_h$.

Corollary 2. *In order to have freedom for optimization, it is required that $n_x > n_h$. If $n_x = n_h$, then only a single point can be feasible, leaving no freedom for optimization, while $n_x < n_h$ violates the assumption of $J_h(x)$ having full row rank.*

To summarize, first-order feasibility for an equality-constrained problem requires that

$$\nabla f(x^*)^\top p \geq 0, \quad J_h(x^*)p = 0 \tag{2.12}$$

For equality constraints, if a direction p is feasible, then a direction $-p$ is also feasible since

$$J_h(x)p = 0 \iff J_h(x)(-p) = 0 \tag{2.13}$$

However, if p is feasible and $\nabla f(x^*)^\top \geq 0$, then $-p$ is also feasible and $\nabla f(x^*)^\top(-p) \leq 0$ which means that a descent direction exists and x^* is not a minimum point.

Corollary 3. *To reach a minimum for an equality-constrained optimization problem, it is required that $\nabla f(x^*)^\top p = 0$.*

Proof. At optimality, it is required that f only increases in the neighborhood of x^*. This can be represented as

$$\nabla f(x^*)^\top p \geq 0, \quad \nabla f(x^*)^\top(-p) \geq 0 \tag{2.14}$$

where p is a feasible direction. (2.14) can be rewritten as

$$\nabla f(x^*)^\top p \geq 0, \quad \nabla f(x^*)^\top p \leq 0 \tag{2.15}$$

which can be further rewritten as

$$0 \leq \nabla f(x^*)^\top p \leq 0 \tag{2.16}$$

[1]Note that there is a slight abuse of notation as $\mathcal{N}(A)$ is used here to represent the null space of a matrix A, but $\mathcal{N}_a$ was used earlier to represent the neighborhood of a point a.

which is only satisfied if

$$\nabla f(x^*)^\top p = 0 \tag{2.17}$$

□

Theorem 2. *At optimality, the gradient of the objective function in an equality-constrained problem must be equal to a linear combination of the gradients of the equality constraints. We can express this as*

$$\nabla f(x^*) = -\sum_{j=1}^{n_h} \lambda_j h_j(x^*) \tag{2.18}$$

*where λ_j are called the **Lagrange multipliers** and a negative sign is used without loss of generality.*

Proof. For x^* to be the minimizer for an equality-constrained problem, it is required that

$$\nabla f(x^*)^\top p = 0, \quad \forall p \in \mathcal{N}(J_h) \tag{2.19}$$

The interpretation behind (2.19) is that the projection of the objective function gradient onto the feasible space must vanish at optimality. We can also express (2.19) as

$$\nabla f \perp \mathcal{N}(J_h) \iff \nabla f \in \mathcal{R}(J_h^\top) \tag{2.20}$$

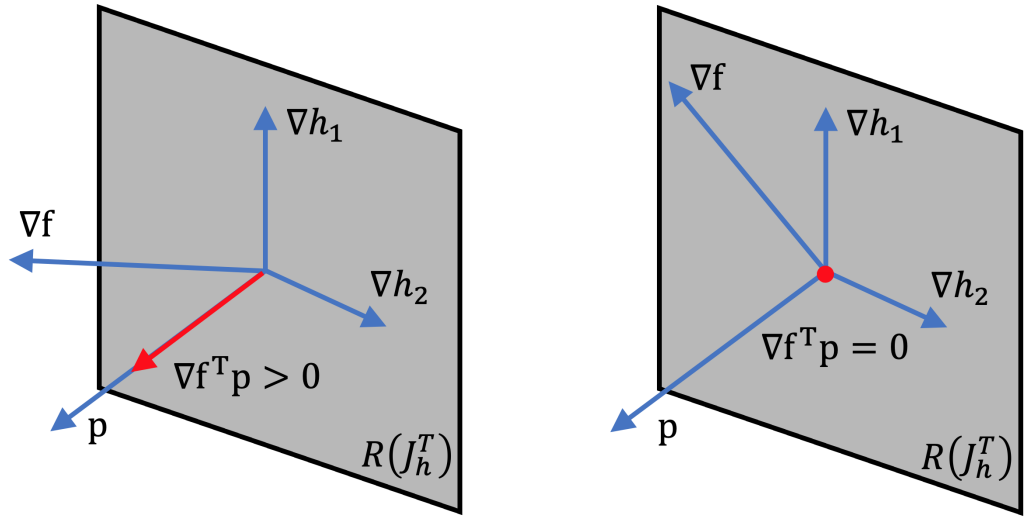
$\mathcal{R}(J_h^\top)$ is the range space, also known as the column space, of $J_h^\top$ which is equivalent to the row space of J_h. The row space of J_h is the span of the gradients of the equality constraints. (2.20) can therefore be interpreted as requiring that the objective function gradient must be a linear combination of the gradients of the equality constraints. This can be visualized in Fig. 2.1 and we can express it as

$$\begin{aligned} \nabla f(x^*) &= -J_h(x^*)^\top \lambda \\ &= -\sum_{j=1}^{n_h} \lambda_j h_j(x^*) \end{aligned} \tag{2.21}$$

□

Remark. *There is a Lagrange multiplier associated with each equality constraint. The sign of the Lagrange multiplier for an equality constraint is arbitrary. However, the sign will be significant later when dealing with inequality constraints.*

Definition. *Combining the equality constraint satisfaction $h(x) = 0$ with (2.21) allows us to write the **first-order necessary optimality conditions for an equality-constrained***



(a) The projection of ∇f onto the feasible space is nonzero, implying that there is a feasible descent direction and therefore the current point is not a minimum.

(b) The projection of ∇f onto the feasible space is zero, implying that there is no feasible descent direction and therefore the current point is a minimum.

Fig. 2.1 Two different scenarios for an equality-constrained problem where $n_x + n_h = 3$.

optimization problem *as*

$$\begin{aligned} \nabla f(x^*) &= -J_h(x^*)^\top \lambda \\ h(x) &= 0 \end{aligned} \tag{2.22}$$

2.2.1 The Lagrangian function for an equality-constrained problem

Definition. *We can define a scalar-valued function for an equality-constrained problem as*

$$\mathcal{L}(x, \lambda) = f(x) + h(x)^\top \lambda \tag{2.23}$$

where the Lagrange multipliers λ are independent variables along with the vector x.

Taking the gradient of $\mathcal{L}$ with respect to x and λ and setting each to 0 gives

$$\nabla_x \mathcal{L} = \nabla f(x) + J_h(x)^\top \lambda = 0 \tag{2.24}$$

$$\nabla_\lambda \mathcal{L} = h(x) = 0 \tag{2.25}$$

which satisfies the first-order necessary optimality conditions defined in (2.22)!

The Lagrangian function allows us to transform a constrained optimization problem into an unconstrained optimization problem by adding new independent variables $\lambda_1, \lambda_2, \ldots, \lambda_{n_h}$. In other words, a constrained problem of n_x design variables and n_h equality constraints is transformed into an unconstrained optimization problem with $n_x + n_h$ variables.

2.2.2 Assumptions for equality-constrained optimization problems

The derivation of the first-order necessary optimality conditions in (2.22) assumes that the gradients of the equality constraints are linearly independent, i.e., J_h has full row rank.

Definition. *A point satisfying the aforementioned linear independence condition is called a* ***regular point*** *and is said to satisfy* ***linear independence constraint qualification*** *(LICQ).*

A special case which does not satisfy LICQ is when one or more constraint gradients are zero. In this case, the constraint gradients are not linearly independent and no points are regular.

2.3 Inequality-Constrained Optimization

An inequality-constrained optimization problem can be written in the following form:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & g(x) \leq 0 \end{aligned}$$

An inequality constraint j is feasible when $g_j(x^*) \leq 0$. Feasible directions for a given inequality constraint reside in a closed half-space.

Definition. *An inequality constraint j is* ***active*** *if $g_j(x^*) = 0$ and* ***inactive*** *if $g_j(x^*) < 0$.*

As in the unconstrained and equality-constrained cases, if x^* is a minimizer for an inequality-constrained problem, any small enough feasible step p from the optimum must result in a function increase. In other words, (2.4) still applies with p needing to respect the inequality constraints. The first-order Taylor series expansion for each inequality constraint can be written as:

$$g_j(x+p) \approx g_j(x) + \nabla g_j(x)^\top p, \quad j = 1, \ldots, n_h \tag{2.27}$$

For a given candidate point which satisfies all constraints, there are two possibilities for each inequality constraint: it is either active or inactive. If a given constraint is inactive, we do not need to enforce any condition on it since we can take a step p in any direction and remain feasible with respect to the inactive constraint, as long as the step is small enough. Therefore, we only need to consider active constraints for the optimality conditions.

From (2.27), if constraint j is active ($g_j(x) = 0$), then the nearby point $g_j(x+p)$ is only feasible if $\nabla g_j(x)^\top p \leq 0$ for all active constraints. This can be written in matrix form as

$$J_g(x)p \leq 0 \tag{2.28}$$

where the Jacobian matrix J_g includes only the gradients of the active inequality constraints. The set of feasible directions, i.e., directions in which all active constraints are satisfied, is all p such that (2.28) is satisfied.

Since any given inequality constraint represents an open half-space, the intersection of all inequality constraints represents a cone. The cone formed by all active inequality constraint gradients is defined by all vectors d such that

$$d = J_g^\top \sigma = \sum_{j=1}^{n_g} \sigma_j \nabla g_j, \quad \text{where } \sigma_j \geq 0 \tag{2.29}$$

This cone is shown in the finely-gridded region in Fig. 2.2.

A direction p is feasible if $p^\top d \leq 0$ for all d in the cone defined in (2.29).

Definition. *The set of all feasible directions forms the* ***polar cone*** *of the cone defined in* (2.29)*. An example of a polar cone is shown in the coarsely-gridded region in Fig. 2.2.*

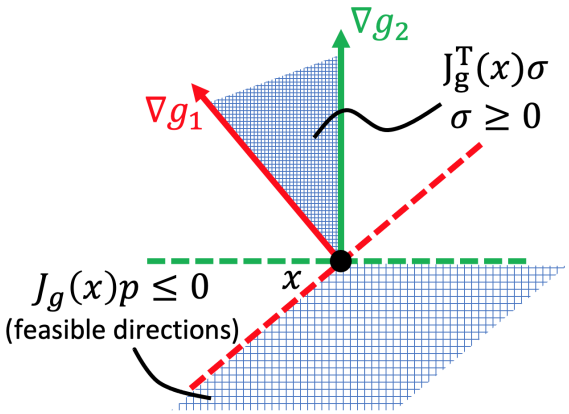
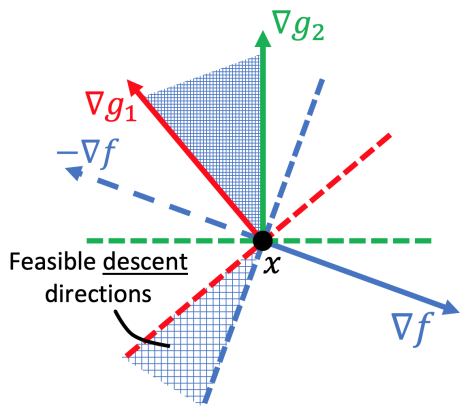


Fig. 2.2 Example of the cone (fine grid) and polar cone (coarse grid) formed by two active inequality constraints, with $x \in \mathbb{R}^2$.

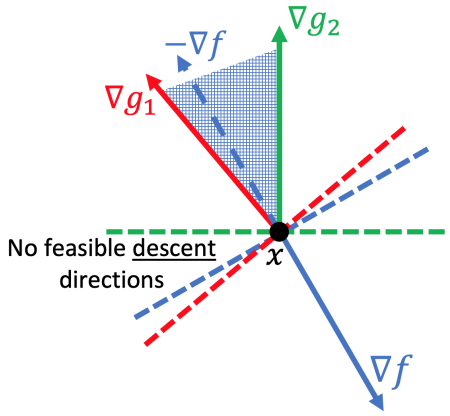
Corollary 4. *An optimum in an inequality-constrained problem has been reached if there is no intersection between the polar cone of feasible directions and the open half-space of descent directions.*

Definition. ***Farkas' lemma*** *states that given a rectangular matrix (J_g in our case) and a vector with the same size as the rows of the matrix (∇f in our case), one and only one of two cases occurs:*

1. *There exists a direction p such that $J_g p \leq 0$ and $\nabla f^\top p \leq 0$. This means that there is a feasible descent direction. This case is shown in Fig. 2.3a.*
2. *There exists a σ such that $J_g^\top \sigma = -\nabla f$ with $\sigma \geq 0$. This corresponds to optimality because it excludes the first case. This second case is shown in Fig. 2.3b.*



(a) The intersection between all possible descent directions and the polar cone of feasible directions is non-empty.



(b) The intersection between all possible descent directions and the polar cone of feasible directions is empty.

Fig. 2.3 Two different scenarios for an inequality-constrained problem where $x \in \mathbb{R}^3$.

The second case in Farkas' Lemma yields the following optimality criterion for inequality constraints:

$$\nabla f(x) + J_g(x)^\top \sigma = 0, \quad \sigma \geq 0 \tag{2.30}$$

Rewriting (2.35) as

$$-\nabla f(x) = J_g(x)^\top \sigma, \quad \sigma \geq 0 \tag{2.31}$$

can perhaps make the optimality criterion easier to visualize using Fig. 2.2 and Fig. 2.3b. In words, it states that the negative of the gradient of the objective function must be a non-negative linear combination (cone) of all active inequality constraint gradients.

Remark. *The requirement in* (2.31) *is similar to* (2.24)*. In* (2.31)*,* σ *corresponds to the Lagrange multipliers for the inequality constraints and carries the additional restriction that* $\sigma \geq 0$*.*

Definition. *Combining the inequality constraint satisfaction* $g(x) \leq 0$ *with* (2.31) *allows us to write the* ***first-order necessary optimality conditions for an inequality-constrained optimization problem*** *as*

$$\begin{aligned} -\nabla f(x^*) &= J_g(x^*)^\top \sigma, \quad \sigma \geq 0 \\ g(x) &\leq 0 \end{aligned} \tag{2.32}$$

2.3.1 The Lagrangian function for an inequality-constrained problem

Similar to the equality-constrained case, we can construct a Lagrangian function whose stationary points are candidates for optimal points. We need to include all inequality constraints in the optimality conditions as we do not know in advance which constraints are

active and which are not. To represent inequality constraints in the Lagrangian, we replace them with equality constrained as such

$$g_j(x) + s_j^2 = 0, \quad j = 1, 2, \ldots, n_g \tag{2.33}$$

where s_j is a new unknown associated with each inequality constraint and called a **slack variable**. Each slack variable in (2.33) is squared to ensure that each s_j^2 term is non-negative and therefore (2.33) can only be satisfied when g_j is feasible ($g_j(x) \leq 0$).

We can define a Lagrangian function for an inequality-constrained problem as

$$\mathcal{L}(x, \sigma, s) = f(x) + \big(g(x) + s \odot s\big)^\top \sigma \tag{2.34}$$

where $s \odot s$ represents the element-wise multiplication of s which is a special case of the Hadamard product of two matrices. The Lagrange multipliers σ and slack variables s are independent variables along with the vector x.

Taking the gradient of $\mathcal{L}$ with respect to the independent variables and setting each to 0 along with adding the constraint $\sigma \geq 0$ gives

$$\nabla_x \mathcal{L} = \nabla f(x) + J_g(x)^\top \sigma = 0 \tag{2.35}$$

$$\nabla_\sigma \mathcal{L} = g_j(x) + s_j^2 = 0, \quad j = 1, 2, \ldots, n_g \tag{2.36}$$

$$\nabla_s \mathcal{L} = 2\sigma_j s_j = 0, \quad j = 1, 2, \ldots, n_g \tag{2.37}$$

$$\sigma \geq 0 \tag{2.38}$$

which satisfies the first-order necessary optimality conditions defined in (2.32)!

Definition. (2.37) *is called the* ***complementary slackness condition****. This condition helps distinguish between active and inactive inequality constraints. For each inequality constraint, either the Lagrange multiplier σ_j is zero (which means that the constraint is inactive), or the slack variable is zero (which means that the constraint is active).*

Remark. *The complementary slackness condition introduces a combinatorial problem. The complexity of it grows exponentially with the number of inequality constraints because the number of possible active versus inactive constraints is 2^{n_g}. There are different techniques for dealing with this. One example, known as an* ***active-set method****, is updating the list of active constraints at each iteration of an optimization algorithm and only solving the resulting equality-constrained problem.*

In addition to the stationary point conditions of the Lagrangian in (2.35)-(2.37), the Lagrange multipliers for the inequality constraints need to be non-negative.

2.3.2 Assumptions for inequality-constrained optimization problems

As in the equality-constrained case, the derivation of the first-order necessary optimality conditions in (2.32) assumes that the gradients of the inequality constraints are linearly independent, i.e., J_g has full row rank.

2.4 Equality- and Inequality-Constrained Optimization

An equality- and inequality-constrained optimization problem can be written in the following form:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & h(x) = 0 \\ & g(x) \leq 0 \end{aligned}$$

Combining ideas from Secs. 2.2 and 2.3, a problem with both equality and inequality constraints can be approached using the following two steps:

1. Apply the same optimality conditions for equality constraints as before
2. Apply the same optimality conditions for inequality constraints as before, but only do so in the subspace of the directions which are feasible with respect to the equality constraints

2.4.1 The Lagrangian function for an equality- and inequality-constrained problem

The Lagrangian for the equality- and inequality-constrained case can be written as[2]

$$\mathcal{L}(x, \lambda, \sigma, s) = f(x) + \lambda^\top h(x) + \sigma^\top \big(g(x) + s \odot s\big) \tag{2.40}$$

Setting the partial derivatives of $\mathcal{L}$ with respect to each unknown to zero yields the following conditions for a stationary point:

$$\begin{aligned} \nabla_x \mathcal{L} &= \nabla f(x) + \lambda^\top J_h(x) + \sigma^\top J_g(x) = 0 \\ \nabla_\lambda \mathcal{L} &= h(x) = 0 \\ \nabla_\sigma \mathcal{L} &= g_j(x) + s_j^2 = 0, \quad j = 1, 2, \ldots, n_g \\ \nabla_s \mathcal{L} &= 2\sigma_j s_j = 0, \quad j = 1, 2, \ldots, n_g \end{aligned} \tag{2.41}$$

[2]Note that some literature flips the meaning of the letters λ and σ, i.e., σ can be used to represent the Lagrange multipliers for equality constraints and λ can be used to represent the Lagrange multipliers for inequality constraints.

In addition to the stationary point conditions of the Lagrangian in (2.41), the Lagrange multipliers for the active constraints ned to be nonnegative (as shown by Farkas' Lemma). Putting all these conditions together in matrix form yields the following **first-order necessary optimality conditions for an equality- and inequality-constrained optimization problem**:

$$\begin{aligned} \nabla f(x) + \lambda^\top J_h(x) + \sigma^\top J_g(x) &= 0 \\ h(x) &= 0 \\ g(x) + s \odot s &= 0 \\ 2\sigma \odot s &= 0 \\ \sigma &\geq 0 \end{aligned} \tag{2.42}$$

2.4.2 Assumptions for equality- and inequality-constrained optimization problems

As in the separate equality- and inequality-constrained cases, the derivation of the first-order necessary optimality conditions in (2.42) assumes that the gradients of the equality constraints and the active inequality constraints are linearly independent, i.e., $\begin{bmatrix} J_h^\top & J_g^\top \end{bmatrix}^\top$ has full row rank.

2.5 KKT Conditions

Definition. *The conditions in* (2.42) *are known as the Karush-Kuhn-Tucker (KKT) conditions.*

We can think of the KKT conditions as a combination of the individual first-order optimality conditions for equality-constrained (2.22) and inequality-constrained (2.32) problems. The KKT conditions have high importance in the field of optimization as they specify the first-order necessary conditions that a point must satisfy in order to be considered a candidate for a minimum. The Lagrangian function $\mathcal{L}$ allows us to take a constrained optimization problem and turn it into an unconstrained optimization problem, making way for a potentially easier path to the solution. A stationary point for the Lagrangian function satisfies the KKT conditions and vice versa. Therefore, we can find the solution to a constrained optimization problem by "simply" finding a point which satisfies the KKT conditions.

Remark. *All discussion up to this point applies to both convex and non-convex optimization problems. However, for a non-convex problem, the ideas discussed apply for finding a local minimum. A local minimum describes the lowest value of the objective function in a small neighborhood around the current point. There are no guarantees regarding if the local minimum*

found for a non-convex problem is also the global minimum. For a convex problem, however, the local minimum is the global minimum. In other words, the size of the neighborhood for which the local minimum holds is unbounded.

Remark. *When a point is not regular, the problem is ill-conditioned which makes it prone for many numerical methods to run into numerical difficulties. The KKT conditions cannot be used with points that are not regular.*

2.6 Second-Order Conditions for Optimality

The optimality conditions described in (2.5), (2.22), and (2.42) are necessary but not sufficient. To make sure that a point is a constrained minimum as opposed to a constrained maximum, we also need to satisfy second-order conditions.

The Lagrangian Hessian is

$$H_{\mathcal{L}} = H_f + \sum_{j=1}^{n_h} \lambda_j H_{h_j} + \sum_{j=1}^{n_g} \sigma_j H_{g_j} \tag{2.43}$$

where H_f is the Hessian of the objective, H_{h_j} is the Hessian of the j-th equality constraint, and H_{g_j} is the Hessian of the j-th inequality constraint. H_f, H_{h_j}, and H_{g_j} can be represented as

$$\begin{aligned}
H_f &= \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & & & \\ \frac{\partial^2 f}{\partial x_1 \partial x_{n_x}} & \frac{\partial^2 f}{\partial x_2 \partial x_{n_x}} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \\
H_{h_j} &= \begin{bmatrix} \frac{\partial^2 h_j}{\partial x_1^2} & \frac{\partial^2 h_j}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 h_j}{\partial x_1 \partial x_n} \\ \vdots & & & \\ \frac{\partial^2 h_j}{\partial x_1 \partial x_{n_x}} & \frac{\partial^2 h_j}{\partial x_2 \partial x_{n_x}} & \cdots & \frac{\partial^2 h_j}{\partial x_n^2} \end{bmatrix} \\
H_{g_j} &= \begin{bmatrix} \frac{\partial^2 g_j}{\partial x_1^2} & \frac{\partial^2 g_j}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 g_j}{\partial x_1 \partial x_n} \\ \vdots & & & \\ \frac{\partial^2 g_j}{\partial x_1 \partial x_{n_x}} & \frac{\partial^2 g_j}{\partial x_2 \partial x_{n_x}} & \cdots & \frac{\partial^2 g_j}{\partial x_n^2} \end{bmatrix}
\end{aligned} \tag{2.44}$$

Definition. *The **second-order sufficient condition** (SOSC) for optimality requires that the curvature is non-negative in any feasible direction.*

For an unconstrained optimization problem, we can represent the SOSC as

$$p^\top H_{\mathcal{L}} p \geq 0 \tag{2.45}$$

For an equality-constrained optimization problem, we can represent the SOSC as

$$p^\top H_{\mathcal{L}} p \geq 0, \quad \forall p \text{ s.t. } J_h p = 0 \tag{2.46}$$

For an inequality-constrained optimization problem, we can represent the SOSC as

$$p^\top H_{\mathcal{L}} p \geq 0, \quad \forall p \text{ s.t. } J_g p \leq 0 \tag{2.47}$$

For an equality- and inequality-constrained optimization problem, we can represent the SOSC as

$$p^\top H_{\mathcal{L}} p \geq 0, \quad \forall p \text{ s.t. } J_h p = 0,\ J_g p \leq 0 \tag{2.48}$$

In words, (2.48) mandates that positive semi-definiteness of the Lagrangian Hessian is only required in the intersection of the nullspace of the equality constraint Jacobian ($\mathcal{N}(J_h)$) with the feasibility cone of the active inequality constraints (polar cone to $J_g^\top \sigma, \sigma \geq 0$).

Definition. ***Strict optimality*** *describes the scenario when there is only one unique minimizer x^* which achieves the minimum value $f(x^*)$. An example where this is possible is the point $x = 1$ for the following problem*

$$\begin{aligned} \min_x \quad & x^2 \\ \textit{subject to} \quad & x \geq 1 \end{aligned}$$

Since any feasible choice for x other than $x = 1$ will yield a greater value for the objective function, $x^* = 1$ is a strict minimizer. An example where strict optimality is not possible is the following problem

$$\begin{aligned} \min_x \quad & xy \\ \textit{subject to} \quad & x \geq 0 \\ & y \geq 0 \end{aligned}$$

Since any value of y can achieve the minimum objective of 0 as long as $x = 0$ and vice versa when $y = 0$, it is not possible to have a unique minimizer.

Remark. *In each of the SOSCs shown in* (2.45)-(2.48)*, we can replace $p^\top H_{\mathcal{L}} p \geq 0$ with $p^\top H_{\mathcal{L}} p > 0$ to mandate* ***strict optimality****.*

2.7 Duality

Definition. *The concept of **duality** in optimization enables us to rewrite an optimization in a different form which is potentially easier to solve. Duality theory states that*

$$p^* = \min_{x,s} \max_{\lambda,\sigma} \mathcal{L}(x, s, \lambda, \sigma) \geq \max_{\lambda,\sigma} \min_{x,s} \mathcal{L}(x, s, \lambda, \sigma) = d^* \tag{2.51}$$

where p is the primal objective and d is the dual objective. d^* gives a lower bound on the primal objective and p^* gives an upper bound on the dual objective.

Definition. *The **duality gap** is calculated as $p^* - d^*$. **Strong duality** describes a situation where $p^* = d^*$. Strong duality is desirable as it allows us to find the minimum of the primal problem directly from the solution of the dual problem.*

Chapter 3

Linear Programming

3.1 Standard Form of an LP

Definition. *A **linear program** (LP) is an optimization problem which can be written in the following standard form*[1]

$$\min_x \quad c^\top x \tag{3.1a}$$

$$\textit{subject to} \quad Ax = b \tag{3.1b}$$

$$Gx \leq h \tag{3.1c}$$

where c and x are vectors in $\mathbb{R}^n$, A is a matrix in $\mathbb{R}^{p\times n}$, b is a vector in $\mathbb{R}^m$, G is a matrix in $\mathbb{R}^{m\times n}$, h is a vector in $\mathbb{R}^p$. The key characteristics of an LP are that the objective is a linear function and the constraints, both equalities and inequalities, are also linear.

3.2 KKT Conditions for a Linear Program

Using (2.42), we can write the KKT conditions for an LP as

$$\begin{aligned} c + A^\top \lambda + G^\top \sigma &= 0 \\ Ax - b &= 0 \\ Gx + s \odot s &= h \\ 2\sigma \odot s &= 0 \\ \sigma &\geq 0 \end{aligned} \tag{3.2}$$

[1] In some literature, the $Gx \leq h$ is replaced with $Gx \geq h$. This is a trivial choice as it can be accommodated by flipping the signs of c, A, b, G, and h. Additionally, some literature replaces $Gx \leq h$ with $x \leq 0$. This can be accomodated by shifting and scaling x.

Remark. *Up until this point, we have used the term $s \odot s$ to replace an inequality constraint with an equality constraint. For example, $Gx \leq h$ is equivalent to $Gx + s \odot s = h$. Some literature, however, instead uses just the extra variable s with the additional constraint $s \geq 0$ to convert an inequality to an equality. An example of this is replacing $Gx \geq h$ with $Gx + s = h$ along with the constraint $s \geq 0$. This no longer has the benefit having a purely equality-constrained problem but certain solution methods are not hindered by this.*

Using $Gx + s = h$ and $s \geq 0$ instead of $Gx + s \odot s = h$, the KKT conditions can be written as[2]

$$c + A^\top \lambda + G^\top \sigma = 0 \tag{3.3a}$$
$$Ax - b = 0 \tag{3.3b}$$
$$Gx + s = h \tag{3.3c}$$
$$\sigma \odot s = 0 \tag{3.3d}$$
$$\sigma \geq 0 \tag{3.3e}$$
$$s \geq 0 \tag{3.3f}$$

Writing (3.3) in matrix-vector form gives

$$\begin{bmatrix} 0 \\ 0 \\ s \end{bmatrix} + \begin{bmatrix} 0 & A^\top & G^\top \\ A & 0 & 0 \\ G & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \\ \sigma \end{bmatrix} = \begin{bmatrix} -c \\ b \\ h \end{bmatrix} \tag{3.4a}$$
$$(s, \sigma) \geq 0 \tag{3.4b}$$
$$s \odot \sigma = 0 \tag{3.4c}$$

Definition. *The following matrix from* (3.4a) *is sometimes referred to as the* ***KKT matrix*** *of a linear program*

$$\begin{bmatrix} 0 & A^\top & G^\top \\ A & 0 & 0 \\ G & 0 & 0 \end{bmatrix} \tag{3.5}$$

If not for the constraints $(s, \sigma) \geq 0$ and $s \odot \sigma = 0$, (3.4) could be written in the form of a system of linear equations for which many known solution methods with low computational complexity exist. However, this is unfortunately not the case and much of the difficulty in solving (3.4) results from the non-negativity constraints on s and σ and the corresponding complementary slackness condition which is nonlinear.

[2] the factor of 2 in $2\sigma \odot s = 0$ has been dropped for simplicity with no impact on the meaning of the KKT conditions.

3.3 Dual of a Linear Program

The dual of an LP can be found through the following steps.

1. Write out the Lagrangian

$$\mathcal{L}(x, s, \lambda, \sigma) = c^\top x + \lambda^\top (Ax - b) + \sigma^\top (Gx + s \odot s - h) \tag{3.6}$$

2. Write out d^*

$$d^* = \max_{\lambda,\sigma} \min_{x,s} c^\top x + \lambda^\top (Ax - b) + \sigma^\top (Gx + s \odot s - h) \tag{3.7}$$

3. Factor out the primal variables x and s

$$d^* = \max_{\lambda,\sigma} \min_{x,s} x^\top (c + A^\top \lambda + G^\top \sigma) - b^\top \lambda - h^\top \sigma + \sigma^\top s \odot s \tag{3.8}$$

4. Observe that the inner minimization yields $-\infty$ unless the following two conditions are met

$$c + A^\top \lambda + G^\top \sigma = 0 \tag{3.9}$$
$$\sigma \geq 0 \tag{3.10}$$

   where (3.10) implies that $\sigma_i s_i = 0$ for $i = 1, 2, \ldots, m$ to achieve the inner minimum of (3.8).

5. If (3.9) and (3.10) are met, then (3.8) reduces to

$$d^* = \max_{\lambda,\sigma} \min_{x,s} -b^\top \lambda - h^\top \sigma \tag{3.11}$$

   after which the terms with x and s are no longer in the expression for d^*, allowing us to remove the inner minimization to obtain

$$d^* = \max_{\lambda,\sigma} -b^\top \lambda - h^\top \sigma \tag{3.12}$$

6. Combining the results from steps 4 and 5 yields

$$\max_{\lambda,\sigma} \quad -b^\top \lambda - h^\top \sigma \tag{3.13a}$$
$$\text{subject to} \quad c + A^\top \lambda + G^\top \sigma = 0 \tag{3.13b}$$
$$\sigma \geq 0 \tag{3.13c}$$

Remark. *The following can be used to transform a maximization problem into a minimization problem and vice versa*

$$\max_x f(x) = \min_x -f(x) \tag{3.14}$$

Given feasible vectors[3] (x, s) for (3.1) and (λ, σ) for (3.13) and applying (3.14) to (3.13), we have from duality theory that

$$c^\top x \geq b^\top \lambda + h^\top \sigma \tag{3.15}$$

[3]For brevity, some abuse of notation is used to write a vector such as $[x^\top, s^\top]^\top$ in the shorter form (x, s).

Chapter 4

Interior-Point Methods

4.1 Introduction

Definition. *A **primal-dual solution** of an LP is the vector $(x^*, s^*, \lambda^*, \sigma^*)$. It contains the solutions to both the primal problem (x^*, s^*) and the dual problem (λ^*, σ^*).*

Definition. ***Interior-point methods** (IPMs) solve optimization problems by iteratively converging towards the optimality (KKT) conditions while keeping the iterates strictly feasible, i.e., $(x^k, s^k, \sigma^k) > 0$ for all k, where k denotes the k-th iteration of an interior-point algorithm.*

Definition. ***Primal-dual interior-point methods** find primal-dual solutions by applying variants of Newton's method to the equality constraints in* (3.4) *and modifying the search direction to satisfy the inequality constraints.*

The KKT conditions in (3.4) can be written in the following alternative form

$$F(x, \lambda, \sigma, s) = \begin{bmatrix} A^\top \lambda + G^\top \sigma + c \\ Ax - b \\ Gx + s - h \\ s \odot \sigma \end{bmatrix} = 0, \qquad (s, \sigma) \geq 0 \tag{4.1}$$

4.2 Newton's Method

Definition. ***Newton's method** uses a linearization of a nonlinear system of equations to determine a search direction in which to move the iterates towards. In the context of a linear program, the linear model used by Newton's method can look as such*

$$J(x, \lambda, \sigma, s) \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \sigma \\ \Delta s \end{bmatrix} = -F(x, \lambda, \sigma, s) \tag{4.2}$$

where J is the Jacobian of F and can be represented as

$$J(x, \lambda, \sigma, s) = \begin{bmatrix} 0 & A^\top & G^\top & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & 0 \\ S & 0 & 0 & X \end{bmatrix} \tag{4.3}$$

If the current iterate is strictly feasible then $F(x, \lambda, \sigma, s) = (0, 0, 0, s \odot \sigma)$ and the Newton step equations become

$$\begin{bmatrix} 0 & A^\top & G^\top & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & 0 \\ S & 0 & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \sigma \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -s \odot \sigma \end{bmatrix} \tag{4.4}$$

A full step is often not permissible due to violation of the strict feasibility requirement for an interior-point method. To avoid this, a line search is used to find the largest step size which still satisfies $(x^k, \sigma^k, s^k) > 0$. The new iterate can be represented in the following form

$$\begin{bmatrix} \Delta x^{k+1} \\ \Delta \lambda^{k+1} \\ \Delta \sigma^{k+1} \\ \Delta s^{k+1} \end{bmatrix} = \alpha \begin{bmatrix} \Delta x^k \\ \Delta \lambda^k \\ \Delta \sigma^k \\ \Delta s^k \end{bmatrix} \tag{4.5}$$

where $\alpha \in [0, 1]$ is the step size found from a line search.

Primal-dual interior-point methods can modify Newton's method in various ways. One common modification is biasing the search direction towards the interior of the feasible set. This allows for making longer step sizes before one of the components of (x^k, σ^k, s^k) becomes non-positive and reduces the likelihood of encountering numerical issues. The search direction is usually biased towards what is called the central path, introduced next.

4.3 Central Path

Definition. *The **central path** is an arc of strictly feasible points parameterized by a scalar τ with each point $(x_\tau, \sigma_\tau, \lambda_\tau, s_\tau)$ solving the following system:*

$$c + A^\top \lambda + G^\top \sigma = 0 \tag{4.6a}$$
$$Ax - b = 0 \tag{4.6b}$$
$$Gx + s = h \tag{4.6c}$$
$$\sigma \odot s = \tau \tag{4.6d}$$
$$\sigma > 0 \tag{4.6e}$$
$$s > 0 \tag{4.6f}$$

These conditions differ from the KKT conditions in (3.3) *only in the term τ on the right-hand side of* (4.6d)*, which replaces the traditional complementary slack condition in* (3.3d)*, and the requirement on σ and s to be strictly positive, however this latter difference is implied in the context of interior-point methods. With the central path in mind, we can also rewrite* (4.1) *as*

$$F(x, \lambda, \sigma, s) = \begin{bmatrix} A^\top \lambda + G^\top \sigma + c \\ Ax - b \\ Gx + s - h \\ s \odot \sigma \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau \end{bmatrix}, \qquad (s, \sigma) \succ 0 \tag{4.7}$$

In brief, the central path can be defined as

$$\mathcal{C} = \{(x_\tau, \lambda_\tau, \sigma_\tau, s_\tau) \mid \tau > 0\} \tag{4.8}$$

Remark. *It can be shown that under certain conditions, the central path is uniquely defined for each $\tau > 0$.*

The importance of the central path is that as τ approaches 0, the system (4.6) approaches the KKT conditions of the linear program in (3.3). If $\mathcal{C}$ converges to anything as $\tau \to 0$, then the solution it converges to is the primal-dual solution of the linear program (3.1).

Definition. *We can define the **duality measure** as*

$$\mu = \frac{1}{m} \sum_{i=1}^{m} s_i \sigma_i = \frac{s^\top \sigma}{m} \tag{4.9}$$

which measures the average of the pairwise products $s_i \sigma_i$ for $i = 1, 2, \ldots, m$.

Definition. *We can next define a **centering parameter** $\phi \in [0, 1]$ which allows us to choose how much we want to bias our search direction towards the central path.*

With consideration of the central path, the Newton step equations become

$$\begin{bmatrix} 0 & A^\top & G^\top & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & 0 \\ S & 0 & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \\ \Delta\sigma \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -s \odot \sigma + \phi\mu\mathbf{e} \end{bmatrix} \tag{4.10}$$

where $\mathbf{e}$ for now is a vector of all ones. This gives a Newton step towards the point $(x_{\phi\mu}, \lambda_{\phi\mu}, \sigma_{\phi\mu}, s_{\phi\mu}) \in \mathcal{C}$ which aims for all the products $s_i\sigma_i$ for $i = 1, 2, \ldots, m$ to be equal to $\phi\mu$. In contrast, the Newton step in (4.4) aims for the point directly satisfying the KKT conditions in (3.3) which is equivalent to choosing $\phi = 0$ in (4.10).

Definition. *Choosing $\phi = 1$ gives what is known as the **centering direction**. The centering direction aims to set all the products $s_i\sigma_i$ for $i = 1, 2, \ldots, m$ equal to μ. Centering directions make little progress, if any, in reducing μ, but they make it easier to make progress on a subsequent step with $\phi < 1$.*

Definition. *Choosing $\phi = 0$ gives the standard Newton step which is sometimes known as the **affine-scaling** direction. This is equivalent to solving directly for the KKT conditions in* (3.3).

Interior-point methods use different approaches to trade off between reducing μ and improving centrality. One of these methods is known as Mehrotra's Predictor Corrector, introduced in Sec. 4.6.

4.4 Infeasible Starting Points

So far, we have assumed that the starting point $(x^0, \lambda^0, \sigma^0, s^0)$ is feasible. In particular, this means that $A^\top\lambda^0 + G^\top\sigma^0 + c = 0$, $Ax^0 = b$, and $Gx^0 + s^0 = h$ are satisfied, and $(x^0, \sigma^0, s^0) > 0$. All subsequent iterates remain strictly feasible due to the zero terms in the right-hand side in (4.10). What if the starting point is not feasible?

Definition. *An interior-point method which does not require for the initial point to be feasible is called an **infeasible interior-point method**. Infeasible interior-point methods only require that x^0 and s^0 are strictly positive.*

For an infeasible interior-point method, the search direction needs to be modified to not only move closer towards centrality but also move closer towards feasibility. This can be achieve through a small modification to the step equation in (4.10). We first define the

residuals for the first three equations in (4.1) as

$$r_x = A^\top \lambda + G^\top \sigma + c \tag{4.11a}$$
$$r_\lambda = Ax - b \tag{4.11b}$$
$$r_\sigma = Gx + s - h \tag{4.11c}$$

Using (4.11), we can define the step equation for an infeasible interior-point method as

$$\begin{bmatrix} 0 & A^\top & G^\top & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & 0 \\ S & 0 & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \sigma \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_x \\ -r_\lambda \\ -r_\sigma \\ -s \odot \sigma + \phi \mu \mathbf{e} \end{bmatrix} \tag{4.12}$$

Remark. *Like the step in* (4.10)*, the step in* (4.12) *is still a Newton step toward the point* $(x_{\phi\mu}, \lambda_{\phi\mu}, \sigma_{\phi\mu}, s_{\phi\mu}) \in \mathcal{C}$*. However, the step in* (4.12) *also tries to remove all infeasibility in the linear equality constraints. If a full step is taken* ($\alpha = 1$)*, then the residuals* r_x*,* r_λ*, and* r_σ *go to zero and all subsequent iterates remain strictly feasible.*

4.5 Primal-Dual Framework

A general framework for a primal-dual interior-point method takes the following form:

Algorithm 1 Primal-Dual Interior-Point Method

Initialize $(x^0, \lambda^0, \sigma^0, s^0)$ such that $(x^0, \sigma^0, s^0) > 0$, e.g., $(1, 0, 1, 1)$

for $k = 1, 2, \ldots$ **do**

Solve

$$\begin{bmatrix} 0 & A^\top & G^\top & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & 0 \\ S^k & 0 & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta \lambda^k \\ \Delta \sigma^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} -r_x^k \\ -r_\lambda^k \\ -r_\sigma^k \\ -s^k \odot \sigma^k + \phi^k \mu^k \mathbf{e} \end{bmatrix}$$

where $\sigma_k \in [0, 1]$ and $\mu_k = \frac{(s^k)^\top \sigma^k}{m}$

Set

$(x^{k+1}, \lambda^{k+1}, \sigma^{k+1}, s^{k+1}) \leftarrow (x^k, \lambda^k, \sigma^k, s^k) + \alpha_k(\Delta x^k, \Delta \lambda^k, \Delta \sigma^k, \Delta s^k)$

where $\alpha_k \in [0, 1]$ is found from a line search to satisfy $(x^{k+1}, \sigma^{k+1}, s^{k+1}) > 0$

end for

4.5.1 Computational Complexity of Interior-Point Methods

Most of the computational complexity in interior-point methods comes from solving linear systems of the form (4.12). The coefficient matrix is usually large and sparse, although the

specific form of it varies from one problem to another and thus influences the choice of matrix factorization to use in solving the linear system. Taking advantage of special structures in the coefficient matrix can allow us to reformulate the step equation in a more compact form which can be solved more quickly than the original form.

4.6 Mehrotra's Predictor-Corrector Algorithm

Although it is desirable for the iterates of an IPM to travel on the central path $\mathcal{C}$ towards the solution, in practice, the iterates are often not feasible and do not lie on the central path. What is of more interest in these cases is a modified trajectory $\mathcal{H}$ that goes from the current iterate and meets up with $\mathcal{C}$ at the solution (or the solution set in the event of multiple minimizers existing).

Algorithms from the framework in Algorithm 1 can be thought of as first-order methods which find the tangent to the path $\mathcal{H}$ at the current iterate and then use a line search to determine how far to move along that tangent. Mehrotra's predictor-corrector algorithm is more advanced as not only does it calculate the same tangent to $\mathcal{H}$ and perform a line search on it, but Mehrotra's algorithm also approximates the curvature of the path $\mathcal{H}$ at the current iterate. This allows Mehrotra's algorithm to have a second-order approximation to the path $\mathcal{H}$. Although this adds some computational complexity, it is often justified in practice as it usually leads to a large reduction in the number of iterations required to achieve optimality.

Many methods predefine the centering parameter ϕ shown in (4.10) or they select it before computing the search direction at each iteration. Mehrotra's predictor-corrector algorithm, on the other hand, adaptively selects the value of ϕ at each iteration before computing the final search direction. At each iteration, Mehrotra's algorithm first calculates the affine-scaling direction ($\phi = 0$). If the affine-scaling direction yields a large reduction in μ without violating the positive constraint $(x, s, \sigma) > 0$, then the algorithm concludes that minimal centering is required and chooses a small ϕ_k close to 0 to calculate the centered search direction. If a large step cannot be taken in the affine-scaling direction, then Mehrotra's algorithm concludes that there is a lot of curvature at the current point on $\mathcal{H}$ and it chooses to use a larger value of ϕ closer to 1 to calculate the centered search direction.

Remark. *Most interior-point algorithms today for general-purpose linear programming are based on Mehrotra's predictor-corrector algorithm.*

4.7 Detecting Infeasibility

The discussion of interior-point method thus far has assumed that a given problem in the form (3.1) is feasible. However, in reality we may stumble across infeasible problems, even after taking measures to avoid such an occurrence. An example of an infeasible problem is

the following:

$$\begin{aligned} \min_{x} \quad & x_1 + x_2 && \text{(4.13a)} \\ \text{subject to} \quad & 3x_1 + 2x_2 = a && \text{(4.13b)} \\ & x_1 \geq 0 && \text{(4.13c)} \\ & x_2 \geq 0 && \text{(4.13d)} \end{aligned}$$

where a is a parameter which can change from one instance of the problem to another. If the parameter a is set to a negative value, problem (4.13) becomes infeasible because constraint (4.13b) cannot be satisfied for $a < 0$ since constraints (4.13c) and (4.13d) imply that $3x_1 + 2x_2 \geq 0$.

A problem can also be unbounded if the objective function is unbounded below in the feasible region. An example of an unbounded problem is the following:

$$\begin{aligned} \min_{x} \quad & x_1 + x_2 && \text{(4.14a)} \\ \text{subject to} \quad & x_1 \leq 5 && \text{(4.14b)} \\ & x_1 \geq 0 && \text{(4.14c)} \\ & x_2 \leq 5 && \text{(4.14d)} \end{aligned}$$

The objective function is unbounded below because as there is no lower bound on the variable x_2 like there is for x_1.

If an algorithm attempts to solve an infeasible or unbounded problem under the assumption that it is feasible and bounded, we might encounter undefined behavior. This is particularly undesirable if we are dependent on the solution to an optimization problem in real-time on a safety-critical autonomous system.

Primal-dual solutions exist whenever the primal and dual problems are both feasible and the solution set is bounded. In such cases, initializing an algorithm with a feasible starting point can help us quickly find the solution. However, it is often not easy to find a feasible starting point. This is where infeasible interior-point methods, as mentioned in Sec. 4.4, come into play. Infeasible IPMs usually begin with a starting point $(x^0, \lambda^0, \sigma^0, s^0)$ that satisfies the strict positivity constraint $(x^0, \sigma^0, s^0) > 0$ but not the equality constraints $A^\top \lambda + G^\top \sigma + c = 0$, $Ax = b$, and $Gx + s = h$. However, perhaps contrary to one's interpretation of the name, an infeasible IPM does not mean it is suited for dealing with a problem that is fundamentally infeasible, i.e., an infeasible IPM can start at an infeasible point but there must still exist a different point which is feasible. What happens if there is not a single feasible point, i.e., the problem itself is infeasible?

It is desirable for an optimization algorithm to be able to perform two key tasks:

- If the problem is feasible: find the solution and return the optimal values of the objective and variables (or return information from which the optimal values can be easily extracted)
- If the problem is infeasible: detect that no solution exists and return this information to the user (ideally with details such as if the infeasibility is in the primal or dual problem)

Additionally, it is desirable that, if the problem is feasible, the algorithm can find the optimal solution even if the starting point is not feasible. One way to achieve most of the above in one algorithm is to embed the given problem into a larger problem for which a strictly feasible point is easy to find. One such example of this is the **Homogeneous Self-Dual Embedding** (HSDE) which will be described in Sec. 4.9 after introducing the concept of self-duality in Sec. 4.8.

4.8 Self-Duality

4.8.1 Linear Complementarity Problems

Definition. *The **monotone linear complementarity problem**, often referred to as an LCP (with the "monotone" being implied), seeks the vectors x and s that satisfy the following conditions:*

$$s = Mx + q, \quad (x, s) \geq 0, \quad x^\top s = 0 \tag{4.15}$$

where M is an $n \times n$ positive semidefinite matrix and q is a vector in $\mathbb{R}^n$. The conditions $x^\top s = 0$ and $(x, s)) \geq 0$ together imply that $x \odot s = 0$.

As written, (4.15) is not an optimization problem since there is nothing to minimize. However, it does have constraints and similarities to the KKT conditions for an LP shown in (3.3). As done so in (3.4), we can rewrite (4.15) in matrix form as

$$F(x, s) = \begin{bmatrix} Mx + q - s \\ x \odot s \end{bmatrix} = 0, \qquad (x, s) \geq 0 \tag{4.16}$$

Definition. *A **mixed monotone linear complementarity problem** (mLCP) seeks the vectors x, s, and z that satisfy the following conditions:*

$$\begin{bmatrix} s \\ 0 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ -M_{12}^\top & M_{22} \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \quad (x, s) \geq 0, \quad x^\top s = 0 \tag{4.17}$$

where the matrix $M \in \mathbb{R}^{(m+n)\times(m+n)}$ *defined by*

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \tag{4.18}$$

is positive semidefinite. Setting $m = 0$*,* $M_{11} = M$*, and* $q_1 = 1$*, it can be see that the LCP in* (4.15) *is a special case of the mLCP in* (4.17)*.*

The KKT conditions for a linear program shown in (3.3) and (3.4) can be expressed in the form of an mLCP as such

$$\begin{bmatrix} s \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -G \\ 0 & 0 & -A \\ G^\top & A^\top & 0 \end{bmatrix} \begin{bmatrix} \sigma \\ \lambda \\ x \end{bmatrix} + \begin{bmatrix} h \\ b \\ c \end{bmatrix}, \quad (s, \sigma) \geq 0, \quad s^\top \sigma = 0 \tag{4.19}$$

where the equivalency to an mLCP is given by the following

$$\begin{gathered} M_{11} = 0, \quad M_{12} = \begin{bmatrix} 0 & -G \end{bmatrix}, \quad M_{21} = \begin{bmatrix} 0 \\ G^\top \end{bmatrix}, \quad M_{22} = \begin{bmatrix} 0 & -A \\ A^\top & 0 \end{bmatrix} \\ x = \sigma, \quad z = \begin{bmatrix} \lambda \\ x \end{bmatrix}, \quad s = s, \quad q_1 = h, \quad q_2 = \begin{bmatrix} b \\ c \end{bmatrix} \end{gathered} \tag{4.20}$$

The step equation for an mLCP, as expressed in (4.17), can be represented as

$$\begin{bmatrix} M_{11} & M_{12} & -I \\ M_{21} & M_{22} & 0 \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta z \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_1 \\ -r_2 \\ -x \odot s + \sigma\mu\mathbf{e} \end{bmatrix} \tag{4.21}$$

4.8.2 Self-Duality of Linear Programs

Definition. *A* ***self-dual*** *linear program has the property that the primal and dual problems are the same problem. This means that solving a self-dual program results in obtaining both the optimal primal variables, e.g.,* x *and* s*, as well as the optimal dual variables, e.g.,* λ *and* σ*.*

In the context of interior-point methods, self-duality enables us to eliminate half the equations from the KKT optimality conditions and therefore solve a problem in a more efficient manner.

A self-dual linear program can be written in the following form:

$$\begin{aligned} \min_{u,w} \quad & f^\top u + g^\top w && \text{(4.22a)} \\ \text{subject to} \quad & M_{11}u + M_{12}w \geq -f && \text{(4.22b)} \\ & -M_{12}^\top u + M_{22}w = -g && \text{(4.22c)} \\ & u \geq 0 && \text{(4.22d)} \end{aligned}$$

where M_{11} and M_{22} are square and skew-symmetric, i.e., $M_{11} = -M_{11}^\top$ and $M_{22} = -M_{22}^\top$. Following the steps described in Sec. 3.3, it can be shown that the dual of (4.22) is simply another copy of (4.22).

It can be shown that the solution to (4.22) can be found from the solution to the following mLCP:

$$\begin{bmatrix} v \\ 0 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ -M_{12}^\top & M_{22} \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix} + \begin{bmatrix} f \\ g \end{bmatrix}, \quad (u, v) \geq 0, \quad u^\top v = 0 \tag{4.23}$$

Any solution (u, v, w) of (4.23) implies a solution (u, w) of (4.22). Likewise, any solution (u, w) of (4.22) implies a solution $(u, M_{11}u + M_{12}w + f, w)$ of (4.23).

4.9 Homogeneous Self-Dual Embedding

4.9.1 Simplified Homogeneous Self-Dual Embedding

The simplified Homogeneous Self-Dual Embedding (HSDE) described by Xu, Hung, and Ye [3] reformulates the original primal and dual problems in (3.1) and (3.13) as a single self-dual linear program in the form of an mLCP which is guaranteed to have a solution. Restating the the original problem and dual problems, we have:

$$\min_{x} \quad c^\top x \quad \text{subject to} \quad Ax = b, \quad Gx \leq h \tag{4.24a}$$

$$\max_{\lambda,\sigma} \quad -b^\top \lambda - h^\top \sigma \quad \text{subject to} \quad c + A^\top \lambda + G^\top \sigma = 0, \quad \sigma \geq 0 \tag{4.24b}$$

The primal and dual problems can be represented as one primal-dual problem in the following manner:

$$\min_{x,\lambda,\sigma} \quad 0 \tag{4.25a}$$

$$\text{subject to} \quad -c^\top x - b^\top \lambda - h^\top \sigma \geq 0 \tag{4.25b}$$

$$h - Gx \geq 0 \tag{4.25c}$$

$$b - Ax = 0 \tag{4.25d}$$

$$c - A^\top \lambda - G^\top \sigma = 0 \tag{4.25e}$$

$$\sigma \geq 0 \tag{4.25f}$$

where (4.25b) comes from duality theory for linear programming, as shown in (3.15), while (4.25c)-(4.25f) come directly from the individual primal and dual problems shown in (4.24).

Introducing a scalar variable τ, we can define the linear program with simplified HSDE as:

$$\min_{x,\lambda,\sigma,\tau} \quad 0 \tag{4.26a}$$

$$\text{subject to} \quad -c^\top x - b^\top \lambda - h^\top \sigma \geq 0 \tag{4.26b}$$

$$h\tau - Gx \geq 0 \tag{4.26c}$$

$$b\tau - Ax = 0 \tag{4.26d}$$

$$c\tau - A^\top \lambda - G^\top \sigma = 0 \tag{4.26e}$$

$$\sigma \geq 0 \tag{4.26f}$$

$$\tau \geq 0 \tag{4.26g}$$

which can be represented in matrix-vector form as:

$$\begin{bmatrix} \kappa \\ s \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & -h^\top & -b^\top & -c^\top \\ h & 0 & 0 & -G \\ b & 0 & 0 & -A \\ c & G^\top & A^\top & 0 \end{bmatrix} \begin{bmatrix} \tau \\ \sigma \\ \lambda \\ x \end{bmatrix}, \quad (\tau, \sigma, \kappa, s) \geq 0, \quad \tau\kappa + s^\top \sigma = 0 \tag{4.27}$$

where κ and s are new variables used to convert the inequality constraints (4.26b) and (4.26c) to equality constraints. The equivalency to an mLCP is given by the following relationships:

$$\begin{gathered} M_{11} = \begin{bmatrix} 0 & -h^\top \\ h & 0 \end{bmatrix}, \quad M_{12} = \begin{bmatrix} -b^\top & -c^\top \\ 0 & -G \end{bmatrix}, \quad M_{21} = \begin{bmatrix} b & 0 \\ c & G^\top \end{bmatrix}, \quad M_{22} = \begin{bmatrix} 0 & -A \\ A^\top & 0 \end{bmatrix} \\ x = \begin{bmatrix} \tau \\ \sigma \end{bmatrix}, \quad z = \begin{bmatrix} \lambda \\ x \end{bmatrix}, \quad s = \begin{bmatrix} \kappa \\ s \end{bmatrix}, \quad q_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad q_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{gathered} \tag{4.28}$$

Furthermore, (4.27) is self-dual as it can be represented in the form shown in (4.23) with the following relationships:

$$\begin{gathered} M_{11} = \begin{bmatrix} 0 & -h^\top \\ h & 0 \end{bmatrix}, \quad M_{12} = \begin{bmatrix} -b^\top & -c^\top \\ 0 & -G \end{bmatrix}, \quad M_{22} = \begin{bmatrix} 0 & -A \\ A^\top & 0 \end{bmatrix} \\ x = \begin{bmatrix} \tau \\ \sigma \end{bmatrix}, \quad z = \begin{bmatrix} \lambda \\ x \end{bmatrix}, \quad v = \begin{bmatrix} \kappa \\ s \end{bmatrix}, \quad f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad g = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{gathered} \tag{4.29}$$

(4.26) is **homogeneous** because the right-hand sides of all the constraints are zero, i.e., there are no independent constants in the constraints. Homogeneity gives the property of having a conical feasible region. If $(x, \lambda, \sigma, \tau)$ is feasible for (4.26), then so is $\alpha(x, \lambda, \sigma, \tau)$ where $\alpha \geq 0$. Since the objective is a constant, each feasible point is a solution to the problem.

To recover information about a solution to the original primal and dual problems in (4.24), a solution $(x^*, \lambda^*, \sigma^*, \tau^*)$ to (4.27) must satisfy the following:

$$\tau^* + \kappa^* > 0, \quad s^* + \sigma^* > 0 \tag{4.30}$$

This is referred to as a **strictly complementary solution** for (4.27).

Theorem 3. *The primal-dual solution set for* (4.24) *is not empty if and only if all strictly complementary solutions of* (4.27) *have* $\tau^* > 0$ *and* $\kappa^* = 0$. *Given such a solution* $(x^*, \lambda^*, \sigma^*, \tau^*, \kappa^*)$ *of* (4.27)*, the point* (x, λ, σ) *defined by*

$$(x, \lambda, \sigma) = \frac{1}{\tau^*}(x^*, \lambda^*, \sigma^*) \tag{4.31}$$

is a primal-dual solution for (4.24)*.*

Proof. See page 181 in [2] for a proof to a similar statement. □

Theorem 4. *If* (4.27) *has a strictly complementary solution* $(x^*, \lambda^*, \sigma^*, \tau^*)$ *for which* $\kappa > 0$*, then at least one of* $c^\top x^*$ *and* $b^\top \lambda + h^\top \sigma$ *is negative and:*

- *if* $c^\top x^* < 0$*, the dual problem* (4.24b) *is infeasible*
- *if* $b^\top \lambda + h^\top \sigma < 0$*, the primal problem* (4.24a) *is infeasible*

Proof. See page 182 in [2] for a proof to a similar statement. □

Remark. *It is not possible to satisfy strict feasibility in* (4.27)*.*

4.9.2 Homogeneous Self-Dual Embedding

The Homogeneous Self-Dual Embedding (HSDE) described by Ye, Todd, and Mizuno [4] incorporates and extra row and column in the constraints compared to the simplified HSDE.

This is done to account for the potential infeasibility of a chosen starting point with respect to the original primal-dual pair in (4.24). The advantage of this embedding is that it becomes easy to choose a strictly feasible starting point. The HSDE of an LP can be expressed in the following form:

$$\min_{x,\lambda,\sigma,\tau,\theta} \quad ((s^0)^\top \sigma^0 + 1)\theta \tag{4.32a}$$

$$\text{subject to} \quad -c^\top x - b^\top \lambda - h^\top \sigma + q_\tau \theta \geq 0 \tag{4.32b}$$

$$h\tau - Gx + q_\sigma \theta \geq 0 \tag{4.32c}$$

$$b\tau - Ax + q_x \theta = 0 \tag{4.32d}$$

$$c\tau - A^\top \lambda - G^\top \sigma + q_\lambda \theta = 0 \tag{4.32e}$$

$$-q_\tau^\top \tau - q_\sigma^\top \sigma - q_\lambda^\top \lambda - q_x^\top x = (s^0)^\top \sigma^0 + 1 \tag{4.32f}$$

$$\sigma \geq 0 \tag{4.32g}$$

$$\tau \geq 0 \tag{4.32h}$$

where

$$\begin{bmatrix} q_\tau \\ q_\sigma \\ q_\lambda \\ q_x \end{bmatrix} = \begin{bmatrix} 1 \\ s^0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 & -h^\top & -b^\top & -c^\top \\ h & 0 & 0 & -G \\ b & 0 & 0 & -A \\ c & G^\top & A^\top & 0 \end{bmatrix} \begin{bmatrix} \tau^0 \\ \sigma^0 \\ \lambda^0 \\ x^0 \end{bmatrix} \tag{4.33}$$

The vectors q_τ, q_σ, q_λ, and q_x represent the infeasibility of the starting point $(\tau^0, \sigma^0, \lambda^0, x^0)$ with respect to the original primal and dual problems as shown in (4.24). The scalar τ has the same purpose as in the simplified HSDE. The nonzero right-hand side in (4.32f) makes (4.32) not strictly homogeneous, although the constraint only exists for the purpose of variable normalization.

(4.32) can be represented in matrix-vector form as:

$$\begin{bmatrix} \kappa \\ s \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & -h^\top & -b^\top & -c^\top & q_\tau \\ h & 0 & 0 & -G & q_\sigma \\ b & 0 & 0 & -A & q_\lambda \\ c & G^\top & A^\top & 0 & q_x \\ -q_\tau^\top & -q_\sigma^\top & -q_\lambda^\top & -q_x^\top & 0 \end{bmatrix} \begin{bmatrix} \tau \\ \sigma \\ \lambda \\ x \\ \theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ (s^0)^\top \sigma^0 + 1 \end{bmatrix}, \quad (\tau, \sigma, \kappa, s) \geq 0,\ \tau\kappa + s^\top \sigma = 0 \tag{4.34}$$

where κ and s are new variables used to convert the inequality constraints (5.4b) and (5.4c) to equality constraints. (4.34) is self-dual as it can be represented in the form shown in (4.23)

with the following relationships:

$$M_{11} = \begin{bmatrix} 0 & -h^\top \\ h & 0 \end{bmatrix}, \quad M_{12} = \begin{bmatrix} -b^\top & -c^\top & q_\tau \\ 0 & -G & q_\sigma \end{bmatrix}, \quad M_{22} = \begin{bmatrix} 0 & -A & q_\lambda \\ A^\top & 0 & q_x \\ -q_\lambda^\top & -q_x^\top & 0 \end{bmatrix}$$
$$x = \begin{bmatrix} \tau \\ \sigma \end{bmatrix}, \quad z = \begin{bmatrix} \lambda \\ x \\ \theta \end{bmatrix}, \quad v = \begin{bmatrix} \kappa \\ s \end{bmatrix}, \quad f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad g = \begin{bmatrix} 0 \\ 0 \\ (s^0)^\top \sigma^0 + 1 \end{bmatrix} \tag{4.35}$$

The mLCP (4.34) is always strictly feasible. A strictly feasible point starting point is given by

$$(x, s, \lambda, \sigma, \tau, \kappa, \theta) = (x^0, s^0, \lambda^0, \sigma^0, 1, 1, 1) \tag{4.36}$$

Likewise, a strictly feasible for (4.32) is given by

$$(x, \lambda, \sigma, \tau, \theta) = (x^0, \lambda^0, \sigma^0, 1, 1) \tag{4.37}$$

Theorem 5. *Any solution* $(x^*, \lambda^*, \sigma^*, \tau^*, \theta^*)$ *of* (4.32) *or* $(x^*, s^*, \lambda^*, \sigma^*, \tau^*, \kappa^*, \theta^*)$ *of* (4.34) *has* $\theta^* = 0$*.*

The property $\theta^* = 0$ means that the last column in (4.34) vanishes at all solutions. This makes the HSDE look like the simplified HSDE. Theorems 3 and 4 can also be extended to the HSDE formulation.

Theorem 6. *The primal-dual solution set for* (4.24) *is not empty if and only if all strictly complementary solutions of* (4.24) *have* $\tau^* > 0$ *and* $\kappa^* = 0$*. Given such a solution* $(x^*, \lambda^*, \sigma^*, \tau^*, \kappa^*, \theta^*)$ *of* (4.34)*, the point* (x, λ, σ) *defined by*

$$(x, \lambda, \sigma) = \frac{1}{\tau^*}(x^*, \lambda^*, \sigma^*) \tag{4.38}$$

is a primal-dual solution for (4.24)*.*

Proof. The proof differs only slightly compared to the one for Theorem 3. □

Theorem 7. *If* (4.34) *has a strictly complementary solution* $(x^*, \lambda^*, \sigma^*, \tau^*)$ *for which* $\kappa > 0$*, then at least one of* $c^\top x^*$ *and* $b^\top \lambda + h^\top \sigma$ *is negative and:*

- *if* $c^\top x^* < 0$*, the dual problem* (4.24b) *is infeasible*
- *if* $b^\top \lambda + h^\top \sigma < 0$*, the primal problem* (4.24a) *is infeasible*

Proof. The proof differs only slightly compared to the one for Theorem 4. □

Chapter 5

Conic Optimization

5.1 Introduction

The discussion in Chapter 4 assumed that the linear program being solved had inequality constraints with respect to a non-negative orthant. An example of such an inequality is $a - b \geq 0$ which implies

$$a_i - b_i \geq 0, \quad i = 1, 2, \ldots, n \tag{5.1}$$

where $a, b \in \mathbb{R}^n$. However, it is also possible to extend the framework from Chapter 4 to account for **generalized inequalities**. A generalized inequality includes **second-order cones**. An example of such an inequality is $s \succeq 0$ which implies

$$s_1 \geq \|s_{2:n}\|_2 \tag{5.2}$$

where $s \in \mathbb{R}^n$.

Using generalized inequalities allows us to cast a variety of convex optimization problems, e.g., linear and quadratic programs, as a **cone program**. A cone program allows for having certain quadratic and higher-order terms in an objective function as well as constraints. A **linear cone program** has a linear objective function but can have generalized inequality constraints along with the standard equality constraints we have already discussed.

In a conic optimization problem, the inequality constraints are not restricted to being just with respect to a non-negative orthant. The inequality constraints can created with respect to a cone.

To preview the HSDE for a conic program, we can choose the following for (4.32)

$$\begin{bmatrix} q_\tau \\ q_\sigma \\ q_\lambda \\ q_x \end{bmatrix} = \frac{m+1}{(s^0)^\top \sigma^0 + 1} \left(\begin{bmatrix} 1 \\ s^0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 & -h^\top & -b^\top & -c^\top \\ h & 0 & 0 & -G \\ b & 0 & 0 & -A \\ c & G^\top & A^\top & 0 \end{bmatrix} \begin{bmatrix} \tau^0 \\ \sigma^0 \\ \lambda^0 \\ x^0 \end{bmatrix} \right) \tag{5.3}$$

where m is the degree of the cone. The new HSDE then becomes

$$\begin{aligned}
\min_{x,\lambda,\sigma,\tau,\theta} \quad & (m+1)\theta && \text{(5.4a)}\\
\text{subject to} \quad & -c^\top x - b^\top \lambda - h^\top \sigma + q_\tau \theta \geq 0 && \text{(5.4b)}\\
& h\tau - Gx + q_\sigma \theta \succeq 0 && \text{(5.4c)}\\
& -b\tau + Ax + q_x \theta = 0 && \text{(5.4d)}\\
& c\tau - A^\top \lambda - G^\top \sigma + q_\lambda \theta = 0 && \text{(5.4e)}\\
& -q_\tau \tau - q_\sigma \sigma - q_\lambda \lambda - q_x x = m+1 && \text{(5.4f)}\\
& \sigma \succeq 0 && \text{(5.4g)}\\
& \tau \geq 0 && \text{(5.4h)}
\end{aligned}$$

which can be represented in matrix-vector form as

$$\begin{bmatrix} \kappa \\ s \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & -h^\top & -b^\top & -c^\top & q_\tau \\ h & 0 & 0 & -G & q_\sigma \\ b & 0 & 0 & -A & q_\lambda \\ c & G^\top & A^\top & 0 & q_x \\ -q_\tau & -q_\sigma & -q_\lambda & -q_x & 0 \end{bmatrix} \begin{bmatrix} \tau \\ \sigma \\ \lambda \\ x \\ \theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ m+1 \end{bmatrix}, \quad (\tau, \sigma, \kappa, s) \succeq 0, \quad \tau\kappa + s^\top \sigma = 0 \tag{5.5}$$

Note how some of the inequalities in (5.4) and (5.5) have changed from a standard linear inequality ($\geq$) to a generalized inequality ($\succeq$).

5.2 Further Reading

Further details on primal-dual interior-point methods with homogeneous self-dual embedding for conic linear programming can be found in [5].

References

[1] Joaquim R. R. A. Martins and Andrew Ning. *Engineering Design Optimization*. Cambridge University Press, Cambridge, UK, January 2022.

[2] Stephen J. Wright. *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics, 1997.

[3] Xiaojie Xu, Pi-Fang Hung, and Yinyu Ye. A simplified homogeneous and self-dual linear programming algorithm and its implementation. *Annals of Operations Research*, 62(1):151–171, Dec 1996.

[4] Yinyu Ye, Michael J. Todd, and Shinji Mizuno. An o($\sqrt{}$nl)-iteration homogeneous and self-dual linear programming algorithm. *Mathematics of Operations Research*, 19(1):53–67, February 1994.

[5] Lieven Vandenberghe. The cvxopt linear and quadratic cone program solvers, March 2010.