

## THE INTERIOR-POINT REVOLUTION IN OPTIMIZATION: HISTORY, RECENT DEVELOPMENTS, AND LASTING CONSEQUENCES

MARGARET H. WRIGHT

ABSTRACT. Interior methods are a pervasive feature of the optimization landscape today, but it was not always so. Although interior-point techniques, primarily in the form of barrier methods, were widely used during the 1960s for problems with nonlinear constraints, their use for the fundamental problem of linear programming was unthinkable because of the total dominance of the simplex method. During the 1970s, barrier methods were superseded, nearly to the point of oblivion, by newly emerging and seemingly more efficient alternatives such as augmented Lagrangian and sequential quadratic programming methods. By the early 1980s, barrier methods were almost universally regarded as a closed chapter in the history of optimization.

This picture changed dramatically in 1984, when Narendra Karmarkar announced a fast polynomial-time interior method for linear programming; in 1985, a formal connection was established between his method and classical barrier methods. Since then, interior methods have continued to transform both the theory and practice of constrained optimization. We present a condensed, unavoidably incomplete look at classical material and recent research about interior methods.

### 1. OVERVIEW

*REVOLUTION:*

- (i) a sudden, radical, or complete change;
- (ii) a fundamental change in political organization, especially the overthrow or renunciation of one government or ruler and the substitution of another.<sup>1</sup>

It can be asserted with a straight face that the field of continuous optimization has undergone a revolution since 1984 in the sense of the first definition and that the second definition applies in a philosophical sense: Because the interior-point presence in optimization today is ubiquitous, it is easy to lose sight of the magnitude and depth of the shifts that have occurred during the past twenty years. Building on the implicit political metaphor of our title, successful revolutions eventually become the status quo.

The interior-point revolution, like many other revolutions, includes old ideas that are rediscovered or seen in a different light, along with genuinely new ideas. The

---

Received by the editors July 9, 2004, and, in revised form, August 17, 2004.

2000 *Mathematics Subject Classification.* Primary 49M37, 65K05, 90C30.

Lecture presented at the AMS Special Session on Current Events, Joint Mathematics Meetings, Phoenix, AZ, January 9, 2004.

<sup>1</sup>*Merriam Webster's Collegiate Dictionary*, Seventh Edition, 1965.

stimulating interplay of old and new continues to lead to increased understanding as well as an ever-larger set of techniques for an ever-larger array of problems, familiar (Section 4.4) and heretofore unexplored (Section 5).

Because of the vast size of the interior-point literature, it would be impractical to cite even a moderate fraction of the relevant references, but more complete treatments are mentioned throughout. The author regrets the impossibility of citing all important work individually.

## 2. LINEAR AND NONLINEAR PROGRAMMING: SEPARATED FROM BIRTH

Prior to 1984, there was, to first order, *no connection* between linear and nonlinear programming. For historical reasons that seem puzzling in retrospect, these topics, one a strict subset of the other, evolved along two essentially disjoint paths. Even more remarkably, this separation was a fully accepted part of the culture of optimization—indeed, it was viewed by some as inherent and unavoidable. For example, in a widely used and highly respected textbook [24] published in 1973, the author comments in the preface that “Part II [unconstrained optimization] . . . is independent of Part I [linear programming]” and that “except in a few isolated sections, this part [constrained optimization] is also independent of Part I.” To provide an accurate reflection of this formerly prevailing viewpoint, we give separate background treatments for linear and nonlinear programming.

### 2.1. Linear programming.

2.1.1. *Problem statement and optimality conditions.* The linear programming (LP) problem involves minimization of a linear (affine) function subject to linear constraints, and can be represented in various mathematically equivalent ways. The two forms of interest here are the all-inequality form,

$$(1) \quad \underset{x}{\text{minimize}} \quad c^T x \quad \text{subject to} \quad Ax \geq b,$$

and *standard form*,

$$(2) \quad \underset{x}{\text{minimize}} \quad c^T x \quad \text{subject to} \quad Ax = b, \quad x \geq 0,$$

where  $A$  is  $m \times n$ . In the standard-form problem (2), the only inequalities are the simple bound constraints  $x \geq 0$ , leading to the crucial (and sometimes overlooked) property that  $x$  plays two distinct roles—as the variables and the values of the constraints. It is customary in standard-form problems to assume that  $A$  has full rank.

A point is *feasible* if it satisfies the problem constraints. The feasible point  $x^*$  is a solution of the standard-form LP (2) if and only if, for some  $m$ -vector  $y^*$  and  $n$ -vector  $z^*$ ,

$$(3) \quad c = A^T y^* + z^*, \quad z^* \geq 0, \quad \text{and} \quad z_i^* x_i^* = 0 \quad \text{for } i = 1, \dots, n,$$

where  $z^*$  is the Lagrange multiplier for the bound constraints and  $y^*$  is the Lagrange multiplier for the equality constraints  $Ax = b$ .

2.1.2. *The simplex method.* A fundamental property of linear programs is that, if the optimal objective value is finite, a *vertex* minimizer must exist. (For details about linear programming and its terminology, see, e.g., [5], [31], and [19].) The simplex method, invented by George B. Dantzig in 1947, is an iterative procedure for solving LPs that completely depends on this property. The starting point for the simplex method must be a vertex. Thereafter, every iteration moves to an adjacent vertex, decreasing the objective as it goes, until an optimal vertex is found. The underlying motivation for the simplex method is easy to understand, but its simplicity is sometimes obscured by a focus on algebraic details.

Almost from the beginning, the simplex method (and, by association, linear programming) acquired an array of specialized terminology and notation, such as “basic feasible solution”, “min ratio test”, and the tableau. During the early years of the simplex method, simplex steps were carried out by performing unsafeguarded rank-one updates to the explicit inverse of the square basis matrix. As an aside, use of this risky technique shows that mainstream linear programming was widely separated not only from nonlinear programming but also from numerical linear algebra; fortunately, during the 1960s, the simplex method became more closely connected with state-of-the-art linear algebraic techniques.

Although “nonsimplex” strategies for LP were suggested and tried from time to time between 1950 and the early 1980s, such techniques never approached the simplex method in overall speed and reliability. Furthermore, the mindset induced by the dominance of the simplex method held sway to such an extent that even techniques labeled as nonsimplex were at heart based on the same motivation as the simplex method: to identify the active inequality constraints by staying on a changing subset of exactly satisfied constraints while reducing the objective function.

2.1.3. *Concerns about complexity.* In practice, because the simplex method routinely and efficiently solved very large linear programs, it retained unquestioned preeminence as the solution method of choice. However, the simplex method was viewed with nagging discontent by those interested in computational complexity, a field whose importance increased during the 1960s and 1970s. An underlying tenet of theoretical computer science is that any “fast” algorithm must be *polynomial-time*, meaning that the number of arithmetic operations required to solve the problem should be bounded above by a polynomial in the problem size.

Although the simplex method almost always converges on real-world problems in a number of iterations that is a small multiple of the problem dimension, it is known that the simplex method can visit every *vertex* of the feasible region—for example, on the famous Klee-Minty “twisted cube” LP; see [31] and [17] for two formulations of this problem. Consequently the worst-case complexity of the simplex method is *exponential* in the problem dimension, which means that the simplex method must be a “bad” algorithm. The disconnect between observed speed and theoretical inefficiency was widely known, and there were several claims, subsequently shown to be false, that a provably polynomial-time method for LP had been discovered.

The first polynomial-time LP algorithm was devised in 1979 by Leonid Khachian of the then Soviet Union, in work that made newspaper headlines around the world. Khachian’s *ellipsoid method* is based on specialization of general *nonlinear* approaches developed earlier by other Soviet mathematicians, notably Shor, Yudin and Nemirovskii. In particular, Khachian’s method does not rely, as the

simplex method does, on existence of a vertex solution or, more generally, the finiteness/combinatorial features of the LP problem. Polynomiality of the ellipsoid method arises from two bounds: an outer bound that guarantees existence of an initial (huge) ellipsoid enclosing the solution and an inner bound that specifies how small the final ellipsoid must be to ensure sufficient closeness to the exact solution. See, for example, [31] for details about Khachian’s method.

Despite its favorable complexity, the performance of the ellipsoid method in practice, i.e., its actual running time, was extremely slow—much slower than the simplex method. In fact, in complete contrast to the simplex method, the number of iterations of the ellipsoid method tended to be comparable to its enormous (albeit polynomial) upper bound. Thus the simplex method remained “the only game in town” for solving linear programs, leading to a puzzling and deeply unsatisfying anomaly in which an exponential-time algorithm was consistently and substantially faster than a polynomial-time algorithm. Even after Khachian’s breakthrough, the quest continued for an LP algorithm that was not only polynomial but also efficient in practice.<sup>2</sup> The linear programming story will continue in Section 3.

## 2.2. Nonlinear programming.

2.2.1. *Problem statement and optimality conditions.* The generic nonlinear programming, or nonlinear optimization, problem involves minimization of a nonlinear function subject to nonlinear constraints. Special cases of nonlinear programming arise when, for example, the objective function is quadratic, the constraints are bounds, or the constraints are linear (equalities or inequalities). Here we consider only the all-inequality version of a nonlinear programming problem:

$$(4) \quad \underset{x \in \mathcal{R}^n}{\text{minimize}} \quad f(x) \quad \text{subject to} \quad c(x) \geq 0,$$

where  $c(x)$  has  $m$  component functions, and  $f$  and  $\{c_i\}$  are smooth. (Observe that (4) is analogous in form to the all-inequality linear program (1).) The  $n$ -vector  $g(x)$  denotes the gradient of  $f$ ; the matrix of second partial derivatives will be denoted by  $H(x)$ . The gradient and Hessian of  $c_i(x)$  will be denoted by  $a_i(x)$  and  $H_i(x)$ . The  $m \times n$  Jacobian matrix of  $c(x)$  is denoted by  $A(x)$ , whose  $i$ th row is  $a_i(x)^T$ . The *Lagrangian function* associated with (4) is  $L(x, \lambda) = f(x) - \lambda^T c(x)$ , where  $\lambda$  normally represents a vector of Lagrange multipliers, one for each constraint. The Hessian of the Lagrangian with respect to  $x$ , denoted by  $W$ , is  $W(x, \lambda) = H - \sum_{j=1}^m \lambda_j H_j(x)$ .

The constraint  $c_i(x) \geq 0$  is said to be *active* at  $\bar{x}$  if  $c_i(\bar{x}) = 0$  and inactive if  $c_i(\bar{x}) > 0$ . Let  $\hat{A}(x)$  denote the Jacobian of the active constraints at  $x$ , and let  $N(x)$  denote a matrix whose columns form a basis for the null space of  $\hat{A}$ .

Throughout the remainder of the paper, we assume the following conditions, which are sufficient to ensure that  $x^*$  is an isolated constrained minimizer of (4):

- (1)  $c(x^*) \geq 0$  and  $\hat{A}(x^*)$  has full rank;
- (2)  $g(x^*) = A(x^*)^T \lambda^*$ , where the optimal Lagrange multiplier  $\lambda^*$  satisfies two conditions:

$$(5) \quad \lambda_j^* \geq 0 \quad \text{and}$$

$$(6) \quad \lambda_j^* c_j(x^*) = 0, \quad j = 1, \dots, m;$$

---

<sup>2</sup> A side comment: Recent work on “smoothed complexity” provides a fascinating explanation of why the simplex method is usually a polynomial-time algorithm; see [32].

- (3)  $\lambda_j^* > 0$  if  $c_j(x^*) = 0$ ,  $j = 1, \dots, m$ ;  
 (4)  $N(x^*)^T W(x^*, \lambda^*) N(x^*)$ , the *reduced Hessian of the Lagrangian*, is positive definite.

Relation (6), that each pairwise product of constraint and multiplier must be zero, is called *complementarity*. Condition 3, *strict complementarity*, requires that one of  $c_j(x^*)$  and  $\lambda_j^*$  must be positive.

2.2.2. *Newton's method.* Newton's method occurs in multiple forms throughout optimization. When solving the nonlinear equations  $\Phi(z) = 0$ , let  $J(z)$  denote the Jacobian matrix of  $\Phi$ . If  $z_k$  is the current point and  $J(z_k)$  is nonsingular, the *Newton step*  $\delta_k$  is the solution of the linear system

$$(7) \quad J(z_k)\delta_k = -\Phi(z_k),$$

so that  $\delta_k$  is the step from  $z_k$  to a zero of the local affine Taylor-series model of  $\Phi$ .

For unconstrained minimization of  $f(x)$  starting from  $x_k$ , the Newton step  $p_k$  is designed to minimize a local Taylor-series quadratic model of  $f(x_k + p)$ , namely  $f(x_k) + g(x_k)^T p + \frac{1}{2} p^T H(x_k) p$ . If the current Hessian  $H(x_k)$  is positive definite,  $p_k$  solves the linear system

$$(8) \quad H(x_k)p = -g(x_k).$$

When minimizing  $f(x)$  subject to  $m$  linear equality constraints  $Ax = b$ , the Newton step  $p_k$  should minimize the local Taylor-series quadratic model of  $f$  subject to also satisfying the constraints  $A(x_k + p_k) = b$ , so that  $p_k$  is a solution of the quadratic program

$$(9) \quad \underset{p \in \mathcal{R}^n}{\text{minimize}} \quad \frac{1}{2} p^T H_k p + g_k^T p \quad \text{subject to} \quad Ap = b - Ax_k,$$

where  $H_k = H(x_k)$  and  $g_k = g(x_k)$ . Under appropriate conditions,  $p_k$  and a “new” multiplier  $y_{k+1}$  satisfy the following  $n + m$  linear equations:

$$(10) \quad \begin{pmatrix} H_k & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} p_k \\ -y_{k+1} \end{pmatrix} = \begin{pmatrix} -g_k \\ b - Ax_k \end{pmatrix},$$

where  $y_{k+1}$  is an estimate of the Lagrange multipliers for the equality constraints. The matrix in (10) is nonsingular if  $A$  has full rank and the reduced Hessian  $N_A^T H_k N_A$  is positive definite, where  $N_A$  is a basis for the null space of  $A$ . If  $Ax_k = b$ , the second equation in (10) becomes  $Ap_k = 0$ , implying that  $p_k$  must lie in the null space of  $A$ .

A “pure” Newton method for zero-finding begins with an initial point  $z_0$  and generates a sequence of Newton iterates  $\{z_k\}$ , where  $z_{k+1} = z_k + \delta_k$ , with  $\delta_k$  defined by (7), and similarly for minimization, using (8) and (10). Under various conditions that can be quite restrictive, a pure Newton method converges quadratically to a solution.

One way to encourage convergence from a general starting point is to perform a *line search* in which the new iterate is defined by  $z_{k+1} = z_k + \alpha_k \delta_k$ , where the positive scalar  $\alpha_k$  is chosen to decrease a *merit function* that measures progress. In unconstrained optimization, the merit function is typically the objective function. Standard line search acceptance criteria that ensure convergence are discussed in, for example, [29, 28]. A second strategy is based on defining a *trust region* around the current iterate within which the local model can be trusted. In optimization, the step in a trust-region method is typically chosen to minimize (approximately)

the local Taylor-series quadratic model subject to remaining within a (normally  $\ell_2$ ) trust region.

**2.2.3. Barrier methods for constrained optimization.** The 1960s were the heyday of unconstrained optimization, and, as a result, it was common practice to convert constrained problems into unconstrained subproblems or sequences of unconstrained subproblems. Penalty and barrier methods were especially popular, both motivated by minimizing a composite function that reflects the original objective function as well as the influence of the constraints. Modern interior methods are closely related to “classical” (1960s) barrier methods, which we now describe.

The *logarithmic barrier function* associated with problem (4) is

$$(11) \quad B(x, \mu) = f(x) - \mu \sum_{j=1}^m \ln c_j(x),$$

where  $\mu$  is a positive scalar called the *barrier parameter*. The logarithmic terms are well defined at points  $x$  for which  $c(x) > 0$ , but become unbounded above as  $x$  approaches any point where a constraint is zero, and are undefined if  $c_j(x) < 0$  for any  $j$ . (This behavior constitutes an obvious rationale for the descriptors “barrier” and “interior”.) Numerous properties of  $B(x, \mu)$  are known; see, for example, the classic reference [9] or [36, 12].

For small  $\mu$ , unconstrained minimizers of  $B(x, \mu)$  are related in an intuitively appealing way to the solution  $x^*$  of (4). Given that  $x^*$  satisfies the sufficient optimality conditions given in Section 2.2.1, then, for a sequence of monotonically decreasing and sufficiently small values of  $\mu$ , there is an associated sequence  $\{x_\mu\}$  of isolated local unconstrained minimizers of the barrier function (11) such that

$$(12) \quad \lim_{\mu \rightarrow 0} x_\mu = x^* \quad \text{and} \quad \lim_{\mu \rightarrow 0} \frac{\mu}{c_j(x_\mu)} = \lambda_j^*.$$

Under suitable assumptions of smoothness, the points  $\{x_\mu\}$  define a smooth curve, called either the *barrier trajectory* or the *central path*, that converges to  $x^*$  *non-tangentially* from the strict interior of the feasible region—not along the boundary. For proofs and additional details, see, for example, [36, 12].

To illustrate the behavior of the log barrier function, we consider the two-variable inequality-constrained problem:

$$(13) \quad \text{minimize} \quad \frac{10}{3}x_1x_2 + \frac{1}{6}x_1$$

$$(14) \quad \text{subject to} \quad \frac{19}{16} - x_1^2 - \frac{5}{2}x_2^2 \geq 0 \quad \text{and} \quad x_1 - x_2 + \frac{3}{5} \geq 0.$$

The first (nonlinear) constraint is satisfied inside an ellipse centered at the origin; the second (linear) constraint cuts off part of the ellipse. Figure 1 shows the contours of  $f$ , which is unbounded below, and the boundaries of these two constraints; the feasible region lies inside the ellipse, to the right of the line.

The figure makes clear that there are two local minimizers of  $f$  in the feasible region. At the isolated constrained minimizer  $x^* = (\frac{3}{4}, \frac{-1}{2})$ , the first constraint is active. The path of barrier minimizers converging to  $x^*$  is shown as a solid line. The strictly feasible starting point of the path of barrier minimizers corresponds to the minimizer of  $-\sum \ln c_i(x)$ —in effect, to an infinite value of  $\mu$ , so that the objective function has no effect.

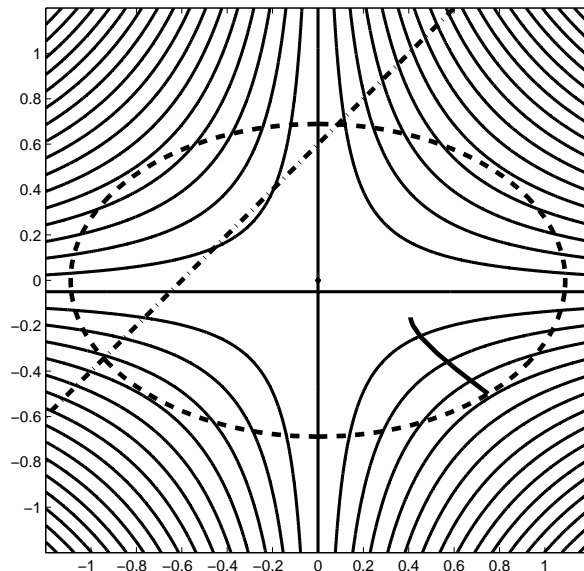


FIGURE 1. The contours of the nonconvex objective function (13) are shown, along with the boundaries of the ellipsoidal constraint (the dashed line) and linear constraint of (14) (the dash-dotted line). A trajectory of local unconstrained minimizers of the logarithmic barrier function, shown as a solid line, begins at the strictly feasible analytic center of the feasible region, corresponding to  $\mu = \infty$ , and converges to the boundary as  $\mu \rightarrow 0$ .

The gradient of the barrier function (11), denoted by  $g_B$ , is

$$(15) \quad g_B(x, \mu) = g(x) - \sum_{j=1}^m \frac{\mu}{c_j(x)} a_j(x) = g(x) - \mu A^T(x) C^{-1}(x) \mathbf{1},$$

where  $\mathbf{1} = (1, \dots, 1)^T$ . The final form in (15) uses the widely established convention in the interior-point literature that an uppercase version of a letter denoting a vector means the diagonal matrix whose diagonal elements are those of the vector. The barrier Hessian, denoted by  $H_B$ , has the form

$$(16) \quad H_B(x, \mu) = H(x) - \sum_{j=1}^m \frac{\mu}{c_j(x)} H_j(x) + \mu A^T(x) C^{-2}(x) A(x).$$

Since  $g_B(x_\mu)$  vanishes,  $x_\mu$  can be interpreted as a highly special point at which the objective gradient is a nonnegative linear combination of the constraint gradients. Further, the coefficients in the linear combination have a specific relationship with  $\mu$  and the constraint values, i.e.,

$$(17) \quad g(x_\mu) = \sum_{j=1}^m \frac{\mu}{c_j(x_\mu)} a_j(x_\mu) = A^T(x_\mu) \lambda_\mu(x_\mu),$$

where the multiplier estimate  $\lambda_\mu$  satisfies

$$(18) \quad \lambda_\mu(x) = \mu C(x)^{-1} \mathbf{1}.$$

A rearranged component-wise version of (18) is

$$(19) \quad (\lambda_\mu(x_\mu))_i c_i(x_\mu) = \mu.$$

This relationship is similar to complementarity (6), which holds at  $x^*$  between  $\lambda^*$  and  $c(x^*)$ , and is sometimes called *perturbed complementarity* or the *centering* property of  $x_\mu$ .

To move from the current point  $x$  to  $x_\mu$ , a straightforward strategy is to apply Newton’s method to a local quadratic model of the barrier function. Omitting arguments, the resulting  $n \times n$  *primal Newton barrier equations* are

$$(20) \quad H_B p = -g + \mu A^T C^{-1} \mathbf{1},$$

where “primal” refers to the original problem variables  $x$ .

Although barrier methods were widely used during the 1960s, they suffered a severe decline in popularity in the 1970s for various reasons, including perceived inefficiency compared to alternative strategies and worries about inherent ill-conditioning. With respect to the latter, it was observed in the late 1960s (see [23, 25]) that, if  $1 \leq \hat{m} < n$ , then  $\text{cond } H_B(x_\mu, \mu) = \Theta(1/\mu)$ , so that the barrier Hessian becomes arbitrarily ill-conditioned at points lying on the barrier trajectory as  $\mu \rightarrow 0$ . Although it is impossible after a gap of more than 20 years to determine precisely why barrier methods became unpopular, concerns about ill-conditioning clearly played a role; see Section 4.3.

As penalty and barrier methods faded from the scene, the dominant approaches tended to be based directly on the optimality conditions for constrained optimization, in particular on properties of the Lagrangian function. Augmented Lagrangian methods and sequential quadratic programming (SQP) methods became especially popular and remain so today. For further details about these methods, see, for example, [16, 10, 28].

### 3. THE REVOLUTION BEGINS

**3.1. Karmarkar’s method.** In 1984, Narendra Karmarkar [21] announced a polynomial-time LP method for which he reported solution times that were consistently 50 times faster than the simplex method. This event, which received publicity around the world throughout the popular press and media, marks the beginning of the interior-point revolution.

Karmarkar’s method had several unusual properties: a special, nonstandard form was assumed for the linear program; nonlinear projective geometry was used in its description; and no information was available about the implementation. Amid the frenzy of interest in Karmarkar’s method, it was shown in 1985 (and published the next year [15]) that there was a formal equivalence between Karmarkar’s method and the classical logarithmic barrier method applied to the LP problem. Soon researchers began to view once-discarded barrier methods in a previously unthinkable context: as the source of polynomial-time algorithms for linear programming.

For several years the tie between Karmarkar’s method and barrier methods was contentious and controversial. Researchers argued about whether the two approaches were fundamentally different, or very similar, or something in between. Now that the dust has settled, derivations of interior methods typically involve barrier functions or their properties, such as perturbed complementarity (19). Readers interested in Karmarkar’s method should consult his original paper [21] or any of the many comprehensive treatments published since 1984 (e.g., [19, 30, 41, 35, 44]).



Beyond establishing the formal connection between Karmarkar's method and barrier methods, [15] reported computational results comparing a state-of-the-art (in 1985) simplex code, MINOS [26], and an implementation of the primal Newton barrier method on a widely available set of test problems. To the astonishment of many who believed that nothing could beat the simplex method, the barrier method was faster on several of the problems and competitive on many others.

At this stage, the interior-point revolution gathered momentum and accelerated in several directions, to be described in Sections 4 and 5. First, however, we describe the derivation of the primal Newton barrier method for LP given in [15].

**3.2. The primal Newton barrier method for LP.** To make the connection between linear programming and a barrier method, consider a standard-form linear program—minimize  $c^T x$  subject to  $Ax = b$  and  $x \geq 0$ —with three properties: (a) the set of  $x$  satisfying  $Ax = b$  and  $x > 0$  is nonempty; (b) the set  $(y, z)$  satisfying  $A^T y + z = c$  and  $z > 0$  is nonempty; and (c)  $\text{rank}(A) = m$ . Because the only inequality constraints are the bounds  $x \geq 0$ , the associated logarithmic barrier function (see (11)) is

$$(21) \quad B(x, \mu) = c^T x - \mu \sum_{j=1}^n \ln x_j,$$

and the barrier subproblem is to minimize (21) subject to satisfying the equalities  $Ax = b$ :

$$(22) \quad \text{minimize } c^T x - \mu \sum_{j=1}^n \ln x_j \quad \text{subject to } Ax = b.$$

The gradient and Hessian of the barrier function (21) have particularly simple forms:

$$(23) \quad g_B = c - \mu X^{-1} e \quad \text{and} \quad H_B = \mu X^{-2}.$$

The barrier subproblem (22) has a unique minimizer if (b) is satisfied. At the optimal solution of (22), there exists  $y$  such that

$$(24) \quad g_B(x, \mu) = c - \mu X^{-1} \mathbf{1} = A^T y, \quad \text{so that } c = A^T y + \mu X^{-1} \mathbf{1}.$$

The *central path* (barrier trajectory) for a standard-form LP is defined by vectors  $x_\mu$  and  $y_\mu$  satisfying

$$(25) \quad \begin{aligned} Ax_\mu &= b, & x_\mu > 0; \\ A^T y_\mu + \mu X_\mu^{-1} \mathbf{1} &= c. \end{aligned}$$

The central path has numerous properties of interest; see, e.g., [20, 19, 35, 41], and [44].

Assume that we are given a point  $x > 0$  for which  $Ax = b$ . Using (23), the Newton equations (10) for problem (22) are

$$(26) \quad \begin{pmatrix} \mu X^{-2} & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} p \\ -y \end{pmatrix} = \begin{pmatrix} -c + \mu X^{-1} \mathbf{1} \\ 0 \end{pmatrix},$$

so that the Newton step  $p$  in  $x$  satisfies

$$(27) \quad \mu X^{-2} p + c - \mu X^{-1} \mathbf{1} = A^T y$$

for some Lagrange multiplier vector  $y$ . Multiplying (27) by  $AX^2$  and using the relation  $Ap = 0$  to eliminate  $p$ , we obtain

$$(28) \quad AX^2A^Ty = AX^2c - \mu AXe = AX(Xc - \mu\mathbf{1}).$$

Since  $A$  has full rank and  $x > 0$ , the matrix  $AX^2A^T$  is positive definite, so that (28) has a unique solution  $y$ . Using (27),  $p$  is defined in terms of  $y$  as

$$p = x + \frac{1}{\mu}X^2(A^Ty - c).$$

Because  $Ax = b$  and  $Ap = 0$ , the new point  $x + \alpha p$  will continue to satisfy the equality constraints for any  $\alpha$ . However,  $\alpha$  may need to be less than one in order to retain strict feasibility with respect to the bound constraints.

#### 4. THE REVOLUTION ADVANCES

Following the announcements of Karmarkar's method and its connection with the logarithmic barrier method, researchers began to develop other interior LP methods with improved complexity bounds and to derive properties of barrier methods applied to linear programs. Furthermore, since barrier methods (unlike the simplex method) were originally intended for nonlinear problems, it was evident that they could be applied not just to linear programming, but also to other optimization problems, such as quadratic programming, linear and nonlinear complementarity, and nonlinear programming.

**4.1. A change in perspective.** The interior-point revolution has led to a fundamental shift in thinking about continuous optimization. Today, in complete contrast to the era before 1984, researchers view linear and nonlinear programming from a *unified* perspective; the magnitude of this change can be seen simply by noting that no one would seriously argue today that linear programming is independent of nonlinear programming.

Beyond a broadened perspective, one might wonder whether the revolution has made a substantive difference: is the net result simply that the log barrier method was rediscovered and applied to new problems? The answer to this is an emphatic "No". As we shall try to indicate in the remainder of the paper, there have been fundamental advances in complexity theory, algorithms, linear algebra, and solvable problems, all as a result of the interior revolution.

**4.2. Complexity.** A signature of interior methods is the existence of continuously parameterized families of approximate solutions that asymptotically converge to the exact solution; see, for example, [20]. As the parameter approaches its limit, these paths trace smooth trajectories with geometric properties (such as being "centered" in a precisely defined sense) that can be analyzed and exploited algorithmically. These paths also play a critical role in complexity analyses of interior algorithms.

The elements in a typical proof of polynomial complexity for an interior method are:

- Characterizing acceptable closeness to the solution through a stopping rule. Such a rule is needed because an interior method that generates strictly feasible iterates cannot produce, within a finite number of iterations, a solution that lies exactly on a constraint.
- Defining a computable measure of closeness to the parameterized path associated with the problem and the algorithm.

- Showing that a Newton step, or a suitably small number of Newton steps, taken from a point close to the path will stay sufficiently close to the path.
- Decreasing the controlling parameter at a rate that allows a polynomial upper bound on the number of iterations needed to become close enough to the solution.

Innumerable papers have been written about complexity issues in interior methods; the surveys [19, 30, 35, 44] (among others) provide details and further references.

Every discussion of the analysis of interior methods should pay tribute to the work of Nesterov and Nemirovskii, whose work in the late 1980s extended the scope of polynomial-time complexity results to a wide family of convex optimization problems; for details, see [27]. One of their major contributions was to define *self-concordant barrier functions*. A convex function  $\phi$  from a convex region  $\mathcal{F}^0 \in \mathcal{R}^n$  to  $\mathcal{R}$  is  $\kappa$ -self-concordant in  $\mathcal{F}^0$  if (i)  $\phi$  is three times continuously differentiable in  $\mathcal{F}^0$  and (ii) for all  $y \in \mathcal{F}^0$  and all  $h \in \mathcal{R}^n$ , the following inequality holds:

$$|\nabla^3\phi(y)[h, h, h]| \leq 2\kappa(h^T\nabla^2\phi(y)h)^{3/2},$$

where  $\nabla^3\phi(y)[h, h, h]$  denotes the third differential of  $\phi$  at  $y$  and  $h$ . The logarithmic barrier functions associated with linear and convex quadratic programming are self-concordant with  $\kappa = 1$ . Existence of a self-concordant barrier function for a convex problem is closely related to existence of polynomial-time algorithms. Using the concept of self-concordance, new barrier functions have been devised for certain convex programming problems, such as semidefinite programming, that were previously considered computationally intractable; see Section 5.

Despite the polynomial bounds typically associated with interior methods, a mystery remains similar to that still holding for the simplex method: interior methods almost invariably require a number of iterations that is much smaller than the (very large) polynomial upper bound. The reasons for these disparities are not yet understood, but perhaps one day they will be.

**4.3. Barrier methods revisited.** The problem of ill-conditioning, as noted earlier, has haunted interior methods since the late 1960s, but there has been substantial recent progress in understanding this issue. A detailed analysis was given in [37] of the structure of the primal barrier Hessian (16) in an entire neighborhood of the solution. Several papers ([13, 11, 40, 42]) have analyzed the stability of specific factorizations for various interior methods.

Very recently, the (at first) surprising result was obtained ([39, 43]) that, under conditions normally holding in practice, ill-conditioning of certain key matrices in interior methods for nonlinear programming does not noticeably degrade the accuracy of the computed search directions. In particular, in modern primal-dual methods (see Section 4.4.1), if a backward-stable method is used to solve the condensed primal-dual system (the analogue of the barrier Hessian), the computed solution has essentially the same accuracy as that of the well-conditioned full primal-dual system (34). However, this result crucially depends on the special structure of the relevant ill-conditioned matrices, in particular their asymptotic relationship with  $\hat{A}$  and the reduced Hessian of the Lagrangian. A similar kind of analysis applies to the search direction computed with the primal barrier Hessian. Consequently, ill-conditioning in interior methods undeniably exists, but will tend to be benign.

It turns out that ill-conditioning is not the only defect of primal barrier methods. Even if the Newton direction is calculated with perfect accuracy, primal barrier

methods suffer from inherently poor scaling of the search direction during the early iterations following a reduction of the barrier parameter; see [38, 6]. Thus, unless special precautions are taken, a full Newton step cannot be taken immediately after the barrier parameter is reduced. This fundamentally undesirable property implies that the classical primal barrier method will be unavoidably inefficient.

A fascinating but unresolvable question is whether the loss of popularity of barrier methods in the 1970s was unfairly blamed on ill-conditioning (which is often a genuine villain); the observed inefficiencies were probably attributable to the just-mentioned flaw of the primal barrier method rather than to ill-conditioning.

**4.4. New algorithms for old problems.** Leading candidates for the most popular algorithms to emerge from the interior revolution belong to the *primal-dual* family. Although there is no precise, universally accepted definition of a primal-dual method, these methods are almost always based on applying Newton's method to nonlinear equations stated in terms of the original ("primal") problem variables, along with "dual" variables representing the Lagrange multipliers.

**4.4.1. Primal-dual methods for linear programming.** The optimal solution  $x$  of the barrier subproblem (22) for a standard-form LP satisfies the condition  $c = A^T y + \mu X^{-1} \mathbf{1}$  for some  $m$ -vector  $y$  (see (24)). Defining the  $n$ -vector  $z$  as  $\mu X^{-1} \mathbf{1}$ , we may replace this condition by the following two equations:

$$(29) \quad c = A^T y + z \quad \text{and} \quad Xz = \mu e.$$

The second relation in (29) has a clear resemblance to the perturbed complementarity condition (19) that holds along the barrier trajectory between the inequality constraints (here, the variables  $x$ ) and Lagrange multiplier estimates.

The primal Newton barrier algorithm described in Section 3.2 is formulated in terms of only primal variables  $x$ ; the Lagrange multiplier estimate  $y$  of (26) arises as a byproduct of the equality-constrained Newton subproblem. One could alternatively seek primal variables  $x$  and dual variables  $y$  (for the equalities) and  $z$  (for the inequalities) satisfying the central-path conditions (25) rewritten to include  $z$ :

$$(30) \quad Ax = b, \quad x > 0, \quad A^T y + z = c, \quad z > 0, \quad \text{and} \quad Xz = \mu \mathbf{1}.$$

Note that only the third equation in (30) is nonlinear.

Applying Newton's method (7) to these  $2n+m$  equations, we obtain the following linear system for Newton steps in  $x$ ,  $y$ , and  $z$ :

$$(31) \quad \begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} = \begin{pmatrix} b - Ax \\ c - A^T y - z \\ \mu e - XZ e \end{pmatrix}.$$

Eliminating  $p_z$  and  $p_x$  gives the linear system

$$(32) \quad AZ^{-1}XA^T p_y = AZ^{-1}X(c - \mu X^{-1}e - A^T y) + b - Ax,$$

where  $AZ^{-1}XA^T$  is symmetric and positive definite, with the form  $AD^2A^T$  for a nonsingular diagonal matrix  $D$ . Once  $p_y$  is known,  $p_z$  and  $p_x$  may be calculated directly from the second and third block rows of (31) without solving any equations.

Primal-dual methods for linear programming have been enormously successful in practice. For a detailed discussion of many aspects of primal-dual methods, see [41].

A striking effect of the interior revolution has been the magnitude and extent of performance improvements in the simplex method, which was (wrongly) thought in 1984 to have already reached its speed limit. In LP today, interior methods are faster than simplex for some very large problems, the reverse is true for some problems, and the two approaches are more or less comparable on others; see [2]. Consequently, commercial LP codes routinely offer both options. Further analysis is still needed of the problem characteristics that determine which approach is more appropriate. Unless a drastic change occurs, both approaches are likely to remain viable into the foreseeable future.

4.4.2. *Primal-dual methods for nonlinear programming.* Because of inherent flaws in primal barrier methods (see Section 4.3), primal-dual methods based on properties of  $x_\mu$  are increasingly popular for solving general nonlinear programming problems; see, for example, the recent papers [8, 4, 11, 7, 14]. As in primal-dual methods for LP, the original (primal) variables  $x$  and the dual variables  $\lambda$  (representing the Lagrange multipliers) are treated as independent.

The usual motivation for primal-dual methods is to find  $(x, \lambda)$  satisfying the equations that hold at  $x_\mu$ . In the spirit of (17) and (18),  $(x_\mu, \lambda_\mu(x_\mu))$  satisfy the following  $n + m$  nonlinear equations:

$$(33) \quad g = A^T \lambda \quad \text{and} \quad c_i \lambda_i = \mu, \quad i = 1, \dots, m.$$

Applying Newton's method, we obtain the (full)  $n + m$  primal-dual equations for Newton steps in  $x$  and  $\lambda$ :

$$(34) \quad \begin{pmatrix} W & -A^T \\ \Lambda A & C \end{pmatrix} \begin{pmatrix} p_x \\ p_\lambda \end{pmatrix} = \begin{pmatrix} -g + A^T \lambda \\ \mu \mathbf{1} - C \lambda \end{pmatrix}$$

where  $W$  is the Hessian of the Lagrangian evaluated at  $(x, \lambda)$ .

All primal-dual methods are based on more or less the idea just described, which is sometimes presented in terms of the logarithmic barrier function (hence leading to properties of  $x_\mu$ ), or else in terms of perturbed complementarity (19) as a desired property in itself. Naturally, the equations (34) do not begin to constitute a complete algorithm for nonlinear programming. Primal-dual methods are the object of active research today and span a wide range of approaches to algorithmic details, including

- (1) formulation of the constraints,
- (2) solution of the linear system that defines the Newton steps,
- (3) treatment of indefiniteness,
- (4) strategies for encouraging progress toward the solution from an arbitrary starting point, and
- (5) treatment of equality constraints (an option needed for a general-purpose nonlinear programming method).

4.5. **Linear algebra.** Interior methods would not be fast or reliable without efficient, numerically stable linear algebraic techniques for solving the associated distinctive, specially structured linear systems. Great advances have taken place since 1984 in sparse Cholesky-based techniques for factorizing matrices of the form  $A^T D^2 A$ , where  $D$  is diagonal and is becoming ill-conditioned in a specified manner—either some elements of  $D$  are becoming infinite while the others are

$\Theta(1)$ , or else some are approaching zero while the remainder are  $\Theta(1)$ . In addition, techniques for sparse symmetric indefinite factorizations of matrices of the form

$$(35) \quad \begin{pmatrix} W & A^T \\ A & D^2 \end{pmatrix},$$

where  $D$  is diagonal and ill-conditioned as just described, are important. See, for example, [11, 13, 40, 42].

## 5. NEW PROBLEMS

The flowering of interior methods, and in particular the realization that efficient algorithms exist for a wide class of convex optimization problems, has led to the application of interior methods to a broad range of problems that were previously considered to be computationally intractable.

Certain problems involving *eigenvalue optimization* have been particularly amenable to solution by interior methods; for details, see the excellent survey [22]. In the next section we summarize a few key ideas in *semidefinite programming* (SDP), an area of intense research during the past few years.

**5.1. The semidefinite programming problem.** Semidefinite programming may be viewed as a generalization of linear programming, where the variables are  $n \times n$  symmetric matrices, denoted by  $X$ , rather than  $n$ -vectors. In SDP, we wish to minimize an affine function of a symmetric matrix  $X$  subject to linear constraints and semidefiniteness constraints, the latter requiring (in words) that “ $X$  must be positive semidefinite”. This relation is typically written as  $X \succeq 0$ , a form that strongly resembles inequality constraints in ordinary continuous optimization. (When  $X$  is a symmetric matrix, the condition  $X \succ 0$  means “ $X$  is positive definite”.)

Let  $\mathcal{S}^n$  denote the set of real  $n \times n$  symmetric matrices, let  $C$  and  $\{A_i\}$  be real symmetric  $n \times n$  matrices, and let  $b$  be a real  $m$ -vector. The semidefinite programming problem is the following:

$$(36) \quad \underset{X \in \mathcal{S}^n}{\text{minimize}} \quad \text{trace}(CX)$$

$$(37) \quad \text{subject to} \quad \text{trace}(A_i X) = b_i, \quad i = 1, \dots, m$$

$$(38) \quad X \succeq 0.$$

When the SDP problem is written in this form, its similarity to a standard-form LP (2) is hard to miss, but, not surprisingly, many extra complications arise in SDP. For example, the feasible region defined by (37) and (38) is not polyhedral, so there is no analogue of the simplex method. Furthermore, several major regularity assumptions are needed to obtain duality results analogous to those in LP. These assumptions will not be stated here; see [22] for details.

Nesterov and Nemirovskii [27] show that the function  $\log \det(X)$  is a self-concordant barrier function for the semidefinite programming problem, which means that the SDP (36)–(38) can be solved in polynomial time via a sequence of barrier subproblems parameterized by  $\mu$ :

$$(39) \quad \underset{X \in \mathcal{S}^n}{\text{minimize}} \quad \text{trace}(CX) - \mu \log \det X$$

$$(40) \quad \text{subject to} \quad \text{trace}(A_i X) = b_i, \quad i = 1, \dots, m.$$

Under suitable regularity assumptions, there is a unique sequence  $\{X_\mu, y_\mu\}$ , where  $X_\mu$  is a symmetric positive definite matrix satisfying the constraints (37)

and (38) and  $y_\mu$  is an  $m$ -vector, such that  $X_\mu$  and  $y_\mu$  together satisfy the following “perturbed complementarity” condition:

$$(41) \quad X(C - \sum_{i=1}^m y_i A_i) = \mu I,$$

with  $C - \sum_{i=1}^m y_i A_i \succeq 0$ . Newton’s method cannot be applied directly to solve (37) and (41) because the matrix on the left-hand side of (41) is not symmetric. A primal approach, first suggested in [1], is to replace (41) by the relation

$$X(C - \sum y_i A_i) + (C - \sum y_i A_i)X = 2\mu I.$$

An analogous primal-dual method, called the “ $XZ + ZX$  method” for obvious reasons, is defined by finding  $(X_\mu, y_\mu, Z_\mu)$ , where  $X_\mu \succ 0$  and  $Z_\mu \succ 0$ , such that

$$(42) \quad \text{trace}(A_i X) = b_i, \quad Z = C - \sum y_i A_i, \quad \text{and} \quad XZ + ZX = 2\mu I.$$

Note the strong parallel between the two final equations in (42) and the primal-dual equations (29) in linear programming.

Semidefinite programming is an extremely lively research area today, producing new theory, algorithms, and implementations; see the surveys [33] and [34].

**5.2. New applications of interior methods.** Interior methods are playing major roles in at least two areas: approximation techniques for NP-hard combinatorial problems, and system and control theory.

In the former, it has recently been shown that certain semidefinite programs and NP-hard problems are closely related in the following way: solution of the semidefinite program leads to an approximation whose objective value is provably within a known factor of the optimal objective value for the associated NP-hard problem. For example, a semidefinite program formulation leads to an approximate solution of the max-cut problem whose objective value is within a factor of 1.14 of the optimal value; see [18]. This kind of relationship guarantees that good approximate solutions to NP-hard problems can be computed in polynomial time.

Interior methods are important in system and control theory because of their connection with *linear matrix inequalities*, which have the forms

$$(43) \quad F_0 + \sum_{i=1}^p x_i F_i \succ 0 \quad \text{or} \quad F_0 + \sum_{i=1}^p x_i F_i \succeq 0,$$

where  $x$  is a  $p$ -vector and  $\{F_i\}$  are real symmetric matrices. Many constraints in system and control theory, including convex quadratic inequalities, matrix norm inequalities, and Lyapunov matrix inequalities, can be expressed as linear matrix inequalities. It is straightforward to see that the forms (43) allow the variables to be symmetric matrices.

Numerous problems in system and control theory involve optimization of convex functions of matrix arguments subject to linear matrix inequalities. Because these are convex programming problems, it is possible to apply polynomial-time interior methods. For details, the reader should consult [3].

## 6. SUMMARY

The interior point revolution has had many highly positive results, including

- a deeper and more unified understanding of constrained optimization problems;
- continuing improvements to theory and methods;
- more algorithmic options for familiar problems, even for linear programming;
- the ability to solve new problems.

One could argue, however, not entirely whimsically, that the interior-point revolution has had some negative consequences. For example, both teaching linear programming and solving linear programs are much more complicated than they used to be. With respect to the former, instructors in linear programming courses face increased pedagogical challenges. Before 1984, it was perfectly acceptable simply to describe the simplex method; today, any credible treatment of linear programming needs to include interior methods. Similarly, someone with an LP to solve can no longer be content with mindless application of the simplex method.

On balance, the interior revolution has energized and expanded the field of constrained optimization. Although the revolutionary pace has (inevitably) slowed down since its first heady days, ample opportunities remain for many further years of lively and innovative research.

## REFERENCES

- [1] F. Alizadeh, J.-P. Haeberly, and M. L. Overton (1998). Primal-dual interior-point methods for semidefinite programming: convergence rates, stability, and numerical results, *SIAM J. Opt.* 8, 746–768. MR1636549 (2000c:90117)
- [2] R. E. Bixby (2002). Solving real-world linear programs: a decade and more of progress, *Operations Research* 50, 3–15. MR1885204
- [3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan (1994). *Linear Matrix Inequalities in System and Control Theory*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania. MR1284712 (95f:93001)
- [4] R. H. Byrd, J. C. Gilbert, and J. Nocedal (2000). A trust region method based on interior point techniques for nonlinear programming, *Math. Prog.* 89, 149–185. MR1795061 (2001j:90120)
- [5] V. Chvátal (1983). *Linear Programming*, W. H. Freeman, New York. MR0717219 (86g:90062)
- [6] A. R. Conn, N. I. M. Gould, and P. L. Toint (1994). A note on using alternative second-order models for the subproblems arising in barrier function methods for minimization, *Num. Math.* 68, 17–33. MR1278446 (95a:90119)
- [7] A. R. Conn, N. I. M. Gould, and P. L. Toint (2000). A primal-dual trust-region algorithm for minimizing a non-convex function subject to bound and linear equality constraints, *Math. Prog.* 87, 215–249. MR1763849 (2001e:90140)
- [8] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang (1996). On the formulation and theory of the Newton interior-point method for nonlinear programming, *J. Opt. Theory Appl.* 89, 507–541. MR1393361 (97c:90104)
- [9] A. V. Fiacco and G. P. McCormick (1968). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley and Sons, New York. Republished by Society for Industrial and Applied Mathematics, Philadelphia, 1990. MR1058438 (91d:90089)
- [10] R. Fletcher (1987). *Practical Methods of Optimization* (second edition), John Wiley and Sons, Chichester. MR0955799 (89j:65050)
- [11] A. Forsgren and P. E. Gill (1998). Primal-dual interior methods for nonconvex nonlinear programming, *SIAM J. Opt.* 8, 1132–1152. MR1646122 (99k:90121)
- [12] A. Forsgren, P. E. Gill, and M. H. Wright (2002). Interior methods for nonlinear optimization, *SIAM Review* 44, 525–597. MR1980444 (2004c:90098)



- [13] A. Forsgren, P. E. Gill, and J. R. Shinnerl (1996). Stability of symmetric ill-conditioned systems arising in interior methods for constrained optimization, *SIAM J. Matrix Anal. Appl.* 17, 187–211. MR1372930 (96m:90084)
- [14] D. M. Gay, M. L. Overton, and M. H. Wright (1998). A primal-dual interior method for nonconvex nonlinear programming, *Advances in Nonlinear Programming* (Y. Yuan, ed.), Kluwer Academic, Dordrecht, 31–56. MR1639869 (99h:90096)
- [15] P. E. Gill, W. Murray, M. A. Saunders, J. A. Tomlin and M. H. Wright (1986). On projected Newton barrier methods for linear programming and an equivalence to Karmarkar’s projective method, *Math. Prog.* 36, 183–209. MR0866988 (88h:90123)
- [16] P. E. Gill, W. Murray and M. H. Wright (1981). *Practical Optimization*, Academic Press, London and New York. MR0634376 (83d:65195)
- [17] P. E. Gill, W. Murray and M. H. Wright (1991). *Numerical Linear Algebra and Optimization, Volume 1*, Addison-Wesley, Redwood City. MR1074004 (92b:65001)
- [18] M. X. Goemans and D. P. Williamson (1995). Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, *J. ACM* 42, 1115–1145. MR1412228 (97g:90108)
- [19] D. Goldfarb and M. J. Todd (1989). Linear programming, *Optimization* (G. L. Nemhauser, A. H. G. Rinnooy Kan and M. J. Todd, eds.), North Holland, Amsterdam and New York, 73–170. MR1105101
- [20] C. C. Gonzaga (1992). Path following methods for linear programming, *SIAM Review* 34, 167–224. MR1166175 (93j:90050)
- [21] N. K. Karmarkar (1984). A new polynomial-time algorithm for linear programming, *Combinatorica* 4, 373–395. MR0779900 (86i:90072)
- [22] A. S. Lewis and M. L. Overton (1996). Eigenvalue optimization, *Acta Numerica 1996*, 149–190. MR1624599 (99e:90072)
- [23] F. A. Lootsma (1969). Hessian matrices of penalty functions for solving constrained optimization problems, *Philips Res. Repts.* 24, 322–330. MR0305594 (46:4724)
- [24] D. G. Luenberger (1973). *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, Menlo Park.
- [25] W. Murray (1971). Analytical expressions for the eigenvalues and eigenvectors of the Hessian matrices of barrier and penalty functions, *J. Opt. Theory Appl.* 7, 189–196. MR0284212 (44:1441)
- [26] B. A. Murtagh and M. A. Saunders (1987). MINOS 5.1 User’s Guide, Report SOL 83-20R, Department of Operations Research, Stanford University, Stanford, California.
- [27] Y. Nesterov and A. Nemirovskii (1994). *Interior-Point Polynomial Algorithms in Convex Programming*, Society for Industrial and Applied Mathematics, Philadelphia. MR1258086 (94m:90005)
- [28] J. Nocedal and S. J. Wright (1999). *Numerical Optimization*, Springer-Verlag, New York. MR1713114 (2001b:90002)
- [29] J. M. Ortega and W. C. Rheinboldt (1970). *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, London and New York. MR0273810 (42:8686)
- [30] C. Roos, T. Terlaky, and J.-Ph. Vial (1997). *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley & Sons, New York. MR1450094 (98d:90005)
- [31] A. Schrijver (1987). *Theory of Linear and Integer Programming*, John Wiley and Sons, New York. MR0874114 (88m:90090)
- [32] D. A. Spielman and S.-H. Teng (2001). Why the simplex method usually takes polynomial time, *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, 296–305.
- [33] M. J. Todd (2001). Semidefinite optimization, *Acta Numerica 2001*, 515–560. MR2009698 (2004g:90004)
- [34] L. Vandenberghe and S. Boyd (1996). Semidefinite programming, *SIAM Review* 38, 49–95. MR1379041 (96m:90005)
- [35] R. J. Vanderbei (1997). *Linear Programming: Foundations and Extensions*, Kluwer Academic Publishers, Boston. MR1845638 (2002e:90002)
- [36] M. H. Wright (1992). Interior methods for constrained optimization, *Acta Numerica 1992*, 341–407. MR1165729 (93d:90037)
- [37] M. H. Wright (1994). Some properties of the Hessian of the logarithmic barrier function, *Math. Prog.* 67, 265–295. MR1305569 (95m:90125)

- [38] M. H. Wright (1995). Why a pure primal Newton barrier step may be infeasible, *SIAM J. Opt.* 5, 1–12. MR1315702 (95m:90126)
- [39] M. H. Wright (1998). Ill-conditioning and computational error in interior methods for non-linear programming, *SIAM J. Opt.* 9, 81–111. MR1660094 (99i:90093)
- [40] S. J. Wright (1995). Stability of linear equation solvers in interior-point methods, *SIAM J. Matrix Anal. Appl.* 16, 1287–1307. MR1351471 (96f:65055)
- [41] S. J. Wright (1997). *Primal-Dual Interior-Point Methods*, Society for Industrial and Applied Mathematics, Philadelphia. MR1422257 (98a:90004)
- [42] S. J. Wright (1999). Modified Cholesky factorizations in interior-point algorithms for linear programming, *SIAM J. Opt.* 9, 1159–1191. MR1724782 (2000k:90076)
- [43] S. J. Wright (2001). Effects of finite-precision arithmetic on interior-point methods for non-linear programming, *SIAM J. Opt.* 12, 36–78. MR1870586 (2002j:90107)
- [44] Y. Ye (1997). *Interior Point Algorithms, Theory and Analysis*, John Wiley & Sons, New York. MR1481160 (98m:90002)

COMPUTER SCIENCE DEPARTMENT, COURANT INSTITUTE OF MATHEMATICAL SCIENCES, NEW YORK UNIVERSITY, 251 MERCER STREET, NEW YORK, NEW YORK 10012  
*E-mail address:* `mhw@cs.nyu.edu`