UC San Diego

UC San Diego Electronic Theses and Dissertations

Title

Projected-Search Methods for Constrained Optimization

Permalink

https://escholarship.org/uc/item/1sj0p4qk

Author

Zhang, Minxin

Publication Date

2023

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

UNIVERSITY OF CALIFORNIA SAN DIEGO

Projected-Search Methods for Constrained Optimization

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Mathematics

by

Minxin Zhang

Committee in charge:

Professor Philip E. Gill, Chair
Professor Randolph E. Bank
Professor Robert R. Bitmead
Professor Michael J. Holst
Professor Wenxin Zhou

2023

Copyright

Minxin Zhang, 2023
All rights reserved.

The Dissertation of Minxin Zhang is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego
2023

DEDICATION

To my family.

EPIGRAPH

A journey of a thousand miles begins with one step.
—Lao Tzu

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Prof. Philip Gill, for his unwavering support in all aspects of my research. His patient guidance, immense passion and exceptional expertise in optimization were essential to the successful completion of this dissertation.

I would also like to extend my appreciation to my committee members: Prof. Randolph Bank, Prof. Robert Bitmead, Prof. Michael Holst and Prof. Wenxin Zhou, for their invaluable support throughout the journey. Special thanks to Prof. Bank and Prof. Holst for the inspiring discussions during the weekly CCoM seminars over the years and for their many other forms of guidance.

Many thanks go to all the friends and colleagues I have been fortunate to share this journey with. A heartfelt thank you to my family, for always offering me their unconditional love and support.

Chapter 3, as well as the numerical results in Sections 5.1–5.2, is partially a reprint of the paper "Projected-search methods for bound-constrained optimization" by Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang, available on arXiv:2110.08359 [math.OC]. Manuscript submitted for publication, 2021. The dissertation author served as the primary investigator and author of the paper. Additionally, Section 3.4 partially reprints the paper by Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang, titled "A limited-memory reduced-Hessian method for bound-constrained optimization." Center for Computational Mathematics Report CCoM 21-01, Center for Computational Mathematics, University of California San Diego, La Jolla, CA, 2021. The dissertation author was the primary author of the content.

Chapter 4 and Section 5.3, in part, reprint the paper by Philip E. Gill and Minxin Zhang, "A projected-search interior method for nonlinear optimization." Manuscript submitted for publication, 2023. The dissertation author was the primary investigator and author of the paper.

Appendix A, in part, reprints the material by Philip E. Gill and Minxin Zhang, "Equations for a Projected-Search Path-Following Method for Nonlinear Optimization." Center for Computational Mathematics Report CCoM 22-02, Center for Computational Mathematics, University of California San Diego, La Jolla, CA, 2022. The dissertation author was the primary author of this material.

VITA

2014	B. S. in Mathematics, Wuhan University
2014-2016	Graduate Research and Teaching Assistant, University of Waterloo
2016	M. Math in Applied Mathematics, University of Waterloo
2017-2023	Graduate Research and Teaching Assistant, University of California San Diego
2023	Ph. D. in Mathematics, University of California San Diego

PUBLICATIONS

Philip E. Gill. and Minxin Zhang. A projected-search interior method for nonlinear optimization. Manuscript submitted for publication, 2023.

Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang. Projected-search methods for bound-constrained optimization. arXiv:2110.08359 [math.OC]. Manuscript submitted for publication, 2021.

Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang. A limited-memory reduced-Hessian method for bound-constrained optimization. Center for Computational Mathematics Report CCoM 21-01, Center for Computational Mathematics, University of California San Diego, La Jolla, CA, 2021.

Minxin Zhang and Kirsten A. Morris. Sensor choice for minimum error variance estimation. IEEE Transactions on Automatic Control, 63(2):315-330, February 2018.

Minxin Zhang. Infinite-dimensional Kalman filtering and sensor placement problem. Master's thesis, University of Waterloo, 2016. Eprint on UWSpace.

Minxin Zhang and Kirsten A. Morris. Effect of sensor noise on estimation of diffusion. *IFAC-PapersOnLine*, Vol.49(8):60-65, 2016. 2nd IFAC Workshop on Control of Systems Governed by Partial Differential Equations, Bertinoro, Italy, 13 -15 June, 2016.

ABSTRACT OF THE DISSERTATION

Projected-Search Methods for Constrained Optimization

by

Minxin Zhang
Doctor of Philosophy in Mathematics

University of California San Diego, 2023

Professor Philip E. Gill, Chair

Projected-search methods for bound-constrained optimization are based on searching along a continuous path obtained by projecting a search direction onto the feasible region. These methods have the potential to change the direction of the search path multiple times along the boundary of the feasible region at the cost of computing a single direction. However, as the objective function is only piecewise differentiable along the path, conventional projected-search methods are limited at using a simple backtracking procedure to obtain a step that satisfies an "Armijo-like" sufficient decrease condition. To extend the benefits of Wolfe line search for unconstrained optimization to projected-search methods, a new quasi-Wolfe step is introduced. Two general classes of projected-search methods that use the new quasi-Wolfe search are then formulated and analyzed. These methods may be broadly categorized as either active-set methods or interior methods. Additionally, a new quasi-Newton projected-search method UBOPT is proposed for unconstrained and bound-constrained optimization. The method computes quasi-Newton directions in a sequence of subspaces, and employs the framework of the class of projected-search active-set methods.

Furthermore, a new interior method is proposed for general nonlinearly constrained optimization, combining a shifted primal-dual interior method with a projected-search method for bound-constrained optimization. The method involves the computation of an approximate Newton direction for a primal-dual penalty-barrier function that incorporates shifts on both the primal and dual variables. The shifts allow the method to be safely “warm started” from a good approximate solution and eliminate the ill-conditioning of the associated linear equations that may occur when the variables are close to zero. The approximate Newton direction is used in conjunction with a new projected-search algorithm that employs a flexible non-monotone quasi-Armijo line search for the minimization of each penalty-barrier function. Numerical results demonstrate that the new method requires significantly fewer iterations than a conventional interior method, thereby reducing the number of times that the search direction need be computed.

Chapter 1

Introduction

1.1 Overview

Generally speaking, optimization is the selection of the best element from the set of all available alternatives, with the goal of maximizing efficiency, effectiveness, or other desired outcomes. The study of optimization relies on the formulation of a mathematical model of a given problem, for which optimizing it means minimizing or maximizing a function, termed the *objective function.* As the maximization of a function is equivalent to the minimization of the negative values of the function, only the minimization needs to be considered here. Unconstrained optimization aims to minimize an objective function $f(x)$ without any constraints on values of x. In contrast, constrained optimization involves a set of constraints that define the acceptable values of the variables. This dissertation focuses on projected-search methods for constrained optimization, with the objective and constraint functions assumed to be twice-continuously differentiable.

One special type of constrained optimization that will be considered is bound-constrained optimization. A bound-constrained problem may be written in the form

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; x \in \mathit{\Omega}, \tag{BC}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is the twice-continuously differentiable objective function and $\mathit{\Omega} = \big\{ x \in \mathbb{R}^n : \ell \le x \le u \big\}$ for vectors of lower and upper bounds such that $\ell \le u$ (with all inequalities defined componentwise). At a given a point x, the active set $\mathcal{A}(x)$ is the set of indices of the variables that lie on their bounds, i.e., $\mathcal{A}(x) = \big\{ i : x_i = \ell_i \;\text{ or }\; x_i = u_i \big\}$.

A general nonlinearly constrained optimization problem may be written in the form

$$\underset{x}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; \begin{pmatrix} \ell^X \\ \ell^S \end{pmatrix} \le \begin{pmatrix} x \\ c(x) \end{pmatrix} \le \begin{pmatrix} u^X \\ u^S \end{pmatrix}, \tag{NP}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is the objective function, $c : \mathbb{R}^n \mapsto \mathbb{R}^m$ represents the vector of constraint functions, and (ℓ^X, ℓ^S) and (u^X, u^S) are constant vectors of lower and upper bounds. In this format, a fixed variable or an equality constraint has the same value for its upper and lower bounds. A variable or constraint with no upper or lower bound is indicated by a bound of $\pm\infty$. By introducing m new variables s, called *slack variables*, the problem (NP) may be reformulated as

$$\underset{x}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) - s = 0, \;\; \begin{pmatrix} \ell^X \\ \ell^S \end{pmatrix} \le \begin{pmatrix} x \\ s \end{pmatrix} \le \begin{pmatrix} u^X \\ u^S \end{pmatrix}. \tag{NPs}$$

In this problem, x and s are treated as independent variables. As the bound constraints $\ell^S \le s \le u^S$ may be reformulated as $s - \ell^S \ge 0$ and $u^S - s \ge 0$, it is usually sufficient to consider a simplified problem of the form

$$\underset{x \in \mathbb{R}^n, s \in \mathbb{R}^m}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) - s = 0, \quad s \ge 0, \tag{NIPs}$$

which is an equivalent form of the inequality-constrained problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) \ge 0. \tag{NIP}$$

A method that is designed to solve (NIPs) may be easily applied to solve the more general problem (NPs) by treating the bound constraints on x in the same way as treating the bound constraints on slack variables s.

All optimization methods discussed in this work are iterative in the following sense. For an optimization problem, if x^* is the exact solution, a sequence $\{\, x_k \,\}$ is generated by an iterative optimization method such that each successive iterate is a new, and ideally, improved estimate of x^*. Although in theory the sequence is infinite, in practice only a finite number of iterates are required to obtain a sufficiently accurate approximation of x^*.

For unconstrained optimization, a line-search method generates a sequence of iterates of the form $x_{k+1} = x_k + \alpha_k p_k$, where p_k is an n-dimensional vector that is called a search direction for f at x_k, and α_k is a positive scalar that is called a step length. Often the step length α_k is chosen to give a decrease in f that is at least as good as a fixed fraction η_A $(0 < \eta_A < \frac{1}{2})$ of the decrease in the local affine model $f(x_k) + \nabla f(x_k)^{\mathrm{T}}(x - x_k)$. If p_k is a descent direction for f at x_k and α_k is a positive scalar, then the decrease condition may be written as

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k \eta_A \nabla f(x_k)^{\mathrm{T}} p_k, \tag{1.1}$$

which is known as the Armijo condition (see Section 2.1.2). Most Armijo line searches are implemented as a simple backtracking procedure in which an initial step is reduced by a constant factor until the Armijo condition (1.1) is satisfied. Alternatively, backtracking may be used in conjunction with a simple quadratic interpolation scheme using $f(x_k)$, $\nabla f(x_k)^{\mathrm{T}} p_k$ and $f(x_k + \alpha p_k)$ at each trial α (see Dennis and Schnabel [21]).

Many practical methods use an α_k that satisfies an additional condition on the directional derivative $\nabla f(x_k + \alpha_k p_k)^{\mathrm{T}} p_k$. In particular, the strong Wolfe conditions require that α_k satisfies both the Armijo condition (1.1) and

$$|\nabla f(x_k + \alpha_k p_k)^{\mathrm{T}} p_k| \leq \eta_W |\nabla f(x_k)^{\mathrm{T}} p_k|, \tag{1.2}$$

where η_W is a preassigned scalar such that $\eta_W \in (\eta_A, 1)$ (see Section 2.1.2). The strong Wolfe conditions allow η_W to be chosen to vary the accuracy of the step. If η_A is fixed at a value close to zero (e.g., 10^{-4}), then a value of η_W close to η_A gives a "tighter" or more accurate step with respect to closeness to a critical point of $\nabla f(x_k + \alpha p_k)^{\mathrm{T}} p_k$. A value of η_W close to one results in a "looser" or more approximate step. A Wolfe line search is able to exploit sophisticated safeguarded polynomial interpolation techniques to provide methods that are more reliable and efficient than those based on backtracking (see, e.g., Hager [60] and Morè and Thuente [71]).

Projected-search methods for bound-constrained optimization can be interpreted as an extension of line-search methods. A projected-search method for problem (BC) generates a sequence of feasible iterates $\{\, x_k \,\}_{k=0}^{\infty}$ such that $x_{k+1} = \mathbf{proj}_{\mathit{\Omega}}(x_k + \alpha_k p_k)$, where p_k is a descent direction for f at x_k, α_k is a scalar step length, and $\mathbf{proj}_{\mathit{\Omega}}(x)$ is the projection of x onto the feasible region, The new iterate may be written as $x_{k+1} = x_k(\alpha_k)$, where $x_k(\alpha)$ denotes the vector $x_k(\alpha) = \mathbf{proj}_{\mathit{\Omega}}(x_k + \alpha p_k)$. A potential benefit of a projected-search method is that many changes to the active set can be made at the cost of computing a single search direction. The projected-search methods of Goldstein [57], Levitin and Polyak [67], and Bertsekas [4] are based on using the gradient-descent direction $p_k = -\nabla f(x_k)$. Bertsekas [6] proposes a method based on computing p_k using a Newton-like method. Calamai and Moré [13] consider methods that identify the optimal active set using a projected-search method and then switch to Newton's method. Projected-search methods based on computing p_k using a quasi-Newton method are proposed by Ni and Yuan [73], Kim, Sra and Dhillon [64], Ferry [28], and Ferry, Gill, Wong and Zhang [30].

In a projected-search method, the function $x_k(\alpha)$ defines a piecewise-linear continuous path, and the function $f\big(x_k(\alpha)\big)$ is not necessarily differentiable along $x_k(\alpha)$. This implies that it is not possible to use a line search based on the conventional Wolfe conditions. Thus, existing projected-search methods are restricted to using a search based on satisfying an Armijo-like condition along the path $x_k(\alpha)$. For the case where $p_k = -\nabla f(x_k)$, a commonly used Armijo-like condition is

$$f\big(x_k(\alpha_k)\big) \le f(x_k) + \eta_A \nabla f(x_k)^{\mathrm{T}}(x_k(\alpha) - x_k), \tag{1.3}$$

proposed by Bertsekas [4] (see also, Calamai and Moré [13]). However, for a general p_k, this may not be a sufficient-decrease condition for a backtracking search as there is no guarantee that the second term on the right-hand side of (1.3) is negative if the path $x_k(\alpha)$ changes direction. An Armijo-like condition that is appropriate for a general descent direction p_k is

$$f\big(x_k(\alpha_k)\big) \le f(x_k) + \alpha_k \eta_A \nabla f(x_k)^{\mathrm{T}} p_k \tag{1.4}$$

(see, e.g., Ni and Yuan [73] and Kim, Sra and Dhillon [64]). Throughout this work, (1.4) is referred to as the *quasi-Armijo* condition. If γ and σ denote fixed parameters such that $\gamma > 0$ and $\sigma \in (0,1)$, then a quasi-Armijo step has the form $\alpha_k = \gamma\sigma^{t_k}$, where t_k is the smallest nonnegative integer such that the quasi-Armijo condition (1.4) is satisfied. Other sufficient decrease conditions have been proposed. For example, Bertsekas [6] considers an Armijo-like condition based on a combination of (1.3) and (1.4), with the term (1.3) defined with components of a scaled gradient-descent direction.

For equality-constrained problems of the form

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) = 0, \tag{NEP}$$

the classical *penalty* methods solve the problem by minimizing a sequence of parameterized penalty functions defined as

$$P_2(x\,; \mu^P) \triangleq f(x) + \frac{1}{2\mu^P}\|c(x)\|^2,$$

where the quadratic penalty term $\frac{1}{2\mu^P}\|c(x)\|^2$ is added to the objective function to penalize the violation of the equality constraints, with the penalty parameter $\mu^P > 0$. The iterates are pushed towards the feasible region by consecutively reducing the penalty parameter μ^P. However, as μ^P approaches zero, the Newton equations associated with minimizing the penalty function become increasingly ill-conditioned. To overcome this ill-conditioning, the augmented Lagrangian method is based on applying the quadratic penalty function to the "shifted" problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\;\; f(x) \quad \text{subject to} \;\; c(x) - \mu^{P} y^{E} = 0, \tag{1.5}$$

where y^{E} represents an estimate of the Lagrangian multiplier vector y. Minimizing the quadratic penalty function for the shifted problem is equivalent to minimizing the augmented Lagrangian function

$$L_{A}(x\,; y^{E}, \mu^{P}) = f(x) - c(x)^{\mathrm{T}} y^{E} + \frac{1}{2\mu^{P}}\|c(x)\|^2,$$

The augmented Lagrangian method enables the iterates to converge to a solution of (NEP) while avoiding the need for μ^{P} to decrease to zero. The method was proposed independently by Hestenes [61] and Powell [79].

To solve a general nonlinearly constrained optimization problem that involves inequality constraints, a popular class of methods is the interior methods. Unlike active-set methods, which seek solutions by moving along the boundary of the feasible region, interior methods work by iteratively moving through the interior of the feasible region. An interior method for the inequality-constrained problem (NIP) is motivated by transforming the original constrained problem into the unconstrained minimization of a sequence of parameterized *barrier* functions. These functions are obtained by combining the objective function with a number of barrier terms that prevent the iterates from violating the inequality constraints.

For a classical barrier method that solves (NIP), reducing the barrier parameter μ^{B} (i.e., the weight of the barrier terms) to zero allows the barrier minimizers to approach a solution of (NIP) from the interior of the feasible region. However, as the barrier parameter and the values of the constraints that are active at the solution approach zero, the linear equations associated with solving each barrier subproblem become increasingly ill-conditioned. Shifted barrier functions were introduced to avoid this ill-conditioning by implicitly shifting the constraint boundary so that the barrier minimizers approach a solution without the need for the barrier parameter to go to zero. This idea was proposed for barrier methods for linear programming by Gill, Murray, Saunders and Wright [52] (see also Freund [41]). Shifted barrier functions are defined in terms of Lagrange multiplier estimates and are analogous to augmented Lagrangian methods for equality-constrained optimization. The advantages of an augmented Lagrangian function over the quadratic penalty function for equality-constrained optimization motivated the class of modified barrier methods, which were proposed independently for nonlinear optimization by Polyak [77] (see Section 2.2.2).

Under certain conditions, the minimizers of the parameterized barrier functions create a continuous path that passes through the solution of (NIP) (see e.g. Theorem 2.2.6). Thus, many interior methods are characterized as path-following methods due to their dependence on properties of such a path. Within the framework of a path-following method, the barrier function may be regarded as a merit function that measures the distance to the path and is used to force convergence of the method. By treating the (primal) variables x and the dual variables y (representing the Lagrangian multiplier) as independent, the primal-dual interior methods are particularly effective in following this path defined by the barrier minimizers (see Section 2.2.3).

Optimization problems with a mixture of equality and inequality constraints may be solved by combining a penalty or augmented Lagrangian method with a barrier method (see Section 2.2.4). In [46], Gill, Kungurtsev and Robinson propose an algorithm for (NIPs) based on using a shifted primal-dual penalty-barrier function as a merit function for a primal-dual path-following method. This function involves a primal-dual shifted penalty term for the equality constraints $c(x) - s = 0$ (see, e.g. Gill and Robinson [53]) and an analogous primal-dual shifted barrier term for the inequalities $s \geq 0$. The method will be reviewed in detail in Section 4.2.

Each penalty-barrier function includes logarithmic terms that generate a singularity at the boundary of the feasible region, implying that the variables are subject to implicit bound constraints during minimization. Conventional interior methods typically minimize the barrier function using unconstrained approaches such as a line-search method, with an artificial upper bound imposed on the step length to prevent the variables from becoming infeasible. However, it is also reasonable to regard it as a bound-constrained problem and solve it utilizing a projected-search method.

1.2 Contributions of This Dissertation

In Chapter 3, two general classes of projected-search methods that employ a new *quasi-Wolfe* line search are formulated for bound-constrained optimization. These methods may be broadly categorized as either active-set methods or interior methods. The new quasi-Wolfe line search, which was initially proposed by Ferry [28], is specifically designed for use with a piecewise linear search path in order to extend the benefits of a conventional Wolfe line search to projected-search methods. Convergence results are established under assumptions that are typical in the analysis of projected-search methods for the class of projected-search active-set methods and the class of projected-search interior methods respectively. In particular, for the active-set methods, it is shown that if the sequence of iterates converges to a nondegenerate stationary point, then the optimal active set can be identified within a finite number of iterations. It follows that once the optimal active set has been identified, any method in this class will have the same convergence rate as its unconstrained counterpart. Additionally, a new quasi-Newton projected-search method UBOPT is proposed in Section 3.4 for solving unconstrained and bound-constrained optimization problems, which is an extension of the limited-memory reduced-Hessian method L-RHR of Leonard [66] and Gill and Leonard [47]. The method UBOPT computes quasi-Newton directions in a sequence of subspaces, and employs the framework of the class of projected-search active-set methods. Convergence properties are established for UBOPT. Numerical results shown in Section 5.1 indicate that using a quasi-Wolfe search in UBOPT resulted in a substantially better performance with respect to function calls and the number of skipped quasi-Newton updates than using a quasi-Armijo search. Numerical results in Section 5.2 indicate that a projected-search interior method for bound-constrained optimization with a quasi-Wolfe line search can provide substantial improvements in robustness and performance compared to a conventional interior method with a Wolfe line search.

Chapter 4 proposes a new projected-search interior method for solving general nonlinear optimization problems of the form (NIPs). As an extension of the shifted primal-dual penalty-barrier method of Gill, Kungurtsev and Robinson [46], a new primal-dual penalty-barrier function is formulated to include shifts on both the dual variables and the slacks. This allows the method to be safely "warm started" from a good approximate solution and eliminates the ill-conditioning of the associated linear equations that may occur when the variables are close to zero. The penalty-barrier function then serves as a merit function for a primal-dual path-following method. An approximated Newton direction is used in conjunction with a novel projected-search algorithm that employs a non-monotone flexible *quasi-Armijo* line search for the minimization of the merit function. Unlike conventional interior methods, projected-search interior methods project the underlying search direction onto a subset of the feasible region defined by perturbing the bounds. Therefore, the direction of the search path may change multiple times along the boundary of the perturbed feasible region at the cost of computing a single direction. The convergence of the projected-search algorithm for minimizing the merit function with fixed parameters is established under certain assumptions. Global convergence results of the new projected-search interior method are also established following a similar procedure as the convergence analysis in Gill, Kungurtsev and Robinson [46]. Numerical results presented in Section 5.3 indicate that the proposed method requires significantly fewer iterations than a conventional interior method, thereby reducing the number of times that a search direction must be computed.

1.3 Notation

Given vectors x and y, the vector consisting of x augmented by y is denoted by (x, y). The subscript i is appended to vectors to denote the ith component of that vector, whereas the subscript k is appended to a vector to denote its value during the kth iteration of an algorithm, e.g., x_k represents the value for x during the kth iteration, whereas $[\, x_k \,]_i$ denotes the ith component of the vector x_k. Given vectors a and b with the same dimension, vectors with ith component $a_i b_i$ and a_i/b_i are denoted by $a \cdot b$ and $a \cdot\!/\, b$ respectively. Given a scalar α, $\alpha \cdot\!/\, b$ is a vector whose ith component is α/b_i. Similarly, $\min(a, b)$ is a vector with components $\min(a_i, b_i)$. The vectors e and e_j denote, respectively, the column vector of ones and the jth column of the identity matrix I. The dimensions of e, e_j and I are defined by the context. The vector two-norm or its induced matrix norm are denoted by $\| \cdot \|$. The inertia of a real symmetric matrix A, denoted by $\mathrm{In}(A)$, is the integer triple (a_+, a_-, a_0) giving the number of positive, negative and zero eigenvalues of A. The n-vector $\nabla f(x)$ denotes gradient of $f(x)$, and the $m \times n$ matrix $J(x)$ denotes the constraint Jacobian, which has ith row $\nabla c_i(x)^{\mathrm{T}}$. Given a Lagrangian function $L(x, y) = f(x) - c(x)^{\mathrm{T}} y$ with y a m-vector of dual variables, the Hessian of the Lagrangian with respect to x is denoted by $H(x, y) = \nabla^2 f(x) - \sum_{i=1}^{m} y_i \nabla^2 c_i(x)$. Let $\{\alpha_j\}_{j \geq 0}$ be a sequence of scalars, vectors, or matrices and let $\{\beta_j\}_{j \geq 0}$ be a sequence of positive scalars. If there exists a positive constant γ such that $\|\alpha_j\| \leq \gamma \beta_j$, we write $\alpha_j = O\big(\beta_j\big)$. If there exists a sequence $\{\gamma_j\} \to 0$ such that $\|\alpha_j\| \leq \gamma_j \beta_j$, we say that $\alpha_j = o(\beta_j)$. If there exists a positive constant γ such that $\|\alpha_j\| > \gamma \beta_j$, we write $\alpha_j = \Omega(\beta_j)$. If there exist positive constants γ_1 and γ_2 such that $\gamma_1 \beta_j \leq \|\alpha_j\| \leq \gamma_2 \beta_j$, we write $\alpha_j = \Theta(\beta_j)$. Given a diagonal matrix $D = \mathrm{diag}(d_1,\, d_2,\, \ldots,\, d_n)$, the Moore-Penrose pseudoinverse of D, denoted by $D^\dagger$, is diagonal with $D^\dagger_{ii} = 0$ for $d_i = 0$ and $D^\dagger_{ii} = 1/d_i$ for $d_i \neq 0$.

Chapter 2

Background

2.1 Line-Search Methods

2.1.1 Fundamentals of unconstrained optimization

Unconstrained optimization focuses on the minimization of a scaler-valued function $f(x)$ without constraints on the values of x. An unconstrained optimization problem may be written in the form

$$\underset{x \in \mathcal{D}}{\text{minimize}} \;\; f(x),$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is the objective function and $\mathcal{D} \subset \mathbb{R}^n$ is the domain of f. It is assumed here that f is twice-continuously differentiable, and $\mathcal{D}$ is an open convex set, e.g., $\mathcal{D} = \mathbb{R}^n$.

A formal definition of a solution of an unconstrained optimization problem is given below.

Definition 2.1.1 (Global unconstrained minimizer)**.** *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, the point $x^* \in \mathcal{D}$ is a* global unconstrained minimizer *of f on $\mathcal{D}$ if $f(x^*) \leq f(x)$ for all $x \in \mathcal{D}$. If x^* is a global unconstrained minimizer, then $f(x^*)$ is called the* global unconstrained minimum *of f.*

Unfortunately, finding a global unconstrained minimizer is computationally intractable except in special cases. Practical methods can be formulated if the conditions on a minimizer are slightly relaxed by focusing on the local behavior of f. Let x^* denote an interior point in $\mathcal{D}$ and define $\mathcal{B}(x^*, \delta)$ as the set of points in $\mathcal{D}$ that lie in an open ball of radius δ centered at x^*, i.e., $\mathcal{B}(x^*, \delta) = \{\, x \in \mathcal{D} : \|x - x^*\| < \delta \,\}$.

Definition 2.1.2 (Local unconstrained minimizer). *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, x^* is a* local unconstrained minimizer *of f if there exists an open ball $\mathcal{B}(x^*, \delta)$ such that $\mathcal{B}(x^*, \delta) \subset \mathcal{D}$ and $f(x^*) \leq f(x)$ for all $x \in \mathcal{B}(x^*, \delta)$.*

Definition 2.1.3 (Strict unconstrained minimizer; weak minimizer). *Given $f\colon \mathcal{D} \subseteq \mathbb{R}^n \mapsto \mathbb{R}$, an unconstrained minimizer x^* of f is a* strict unconstrained minimizer *if there exists a neighborhood $\mathcal{B}(x^*, \delta) \subset \mathcal{D}$ such that*

$$f(x^*) < f(x) \quad \text{for all } x \in \mathcal{B}(x^*, \delta),\ x \neq x^*.$$

An unconstrained minimizer x^* is a weak *unconstrained minimizer if it is not a strict unconstrained minimizer.*

Definition 2.1.4 (Isolated unconstrained minimizer). *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, an unconstrained minimizer x^* is an* isolated unconstrained minimizer *of f if there exists an open ball $\mathcal{B}(x^*, \delta)$ such that x^* is the only unconstrained minimizer in $\mathcal{B}(x^*, \delta)$.*

Theorem 2.1.1 (First-order necessary condition for an unconstrained minimizer). *Given $f : \mathbb{R}^n \mapsto \mathbb{R}$, assume that x^* is an unconstrained minimizer of f, and that f is differentiable at x^*. Then $\nabla f(x^*) = 0$.*

Proof. By assumption, x^* is an unconstrained minimizer, which implies that

$$f(x^* + tp) \geq f(x^*) \ \text{ for all } p \in \mathbb{R}^n \text{ and, given } p, \text{ all sufficiently small } t. \tag{2.1}$$

Because f is differentiable at x^*, the directional derivative of f along any direction p is $\nabla f(x^*)^{\mathrm{T}} p$, and if t approaches zero from the positive side,

$$\lim_{t \to 0+} \frac{1}{t}\big(f(x^* + tp) - f(x^*)\big) = \nabla f(x^*)^{\mathrm{T}} p \ \text{ for all } p \in \mathbb{R}^n. \tag{2.2}$$

The combination of (2.1) and (2.2) implies that $\nabla f(x^*)^{\mathrm{T}} p \geq 0$ for all $p \in \mathbb{R}^n$, which can be true only if $\nabla f(x^*) = 0$. □

Definition 2.1.5 (Unconstrained stationary point). *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, assume that x^* is an interior point of $\mathcal{D}$ and that f is differentiable at x^*. If $\nabla f(x^*) = 0$, x^* is called a* stationary point *of f.*

Theorem 2.1.2 (Second-order necessary conditions for an unconstrained minimizer [54]). *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, assume that x^* is an unconstrained minimizer of f, and that the second-*

order derivative of f exists at x^*. Then $\nabla f(x^*) = 0$ and the Hessian matrix $\nabla^2 f(x^*)$ is positive semidefinite.

Proof. As x^* is an unconstrained minimizer, it must be an interior point of $\mathcal{D}$. Existence of the second derivative of f at x^* implies, that the first derivative exists. The fact that $\nabla f(x^*) = 0$ under these circumstances follows from Theorem 2.1.1.

We now use contradiction to prove that $\nabla^2 f(x^*)$ is positive semidefinite. Because the second derivative of f exists at x^*, a second-order Taylor-series expansion of $f(x)$ at x^* implies that

$$\lim_{t\to 0} \frac{1}{t^2}\bigl[f(x^* + tp) - f(x^*)\bigr] = \tfrac{1}{2} p^{\mathrm{T}} \nabla^2 f(x^*) p \quad \text{for all } p \in \mathbb{R}^n. \tag{2.3}$$

If the Hessian matrix $\nabla^2 f(x^*)$ is not positive semidefinite, then there is at least one vector $\hat{p}$ such that $\hat{p}^{\mathrm{T}} \nabla^2 f(x^*) \hat{p} < 0$. For $p = \hat{p}$, it follows from (2.3) that

$$\lim_{t\to 0} \frac{1}{t^2}\bigl[f(x^* + t\hat{p}) - f(x^*)\bigr] < 0. \tag{2.4}$$

The fact that x^* is an unconstrained minimizer means that, according to Definition 2.1.2, $f(x^* + tp) - f(x^*) \geq 0$ for *all* p and sufficiently small t. But relation (2.4) implies that, for sufficiently small t, $f(x^* + t\hat{p}) - f(x^*) < 0$, a contradiction that gives the desired result. □

Theorem 2.1.3 (Sufficient conditions for an isolated minimizer [54])**.** *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, assume that x^* is an unconstrained minimizer of f, and that the second-order derivative of f exists at x^*. If $\nabla f(x^*) = 0$ and the Hessian matrix $\nabla^2 f(x^*)$ is positive definite, then x^* is an isolated unconstrained minimizer.*

Proof. First we show that x^* is a strict minimizer. By assumption $\nabla f(x^*) = 0$ and the second derivative of f exists at x^*, so that we have

$$\lim_{t\to 0} \frac{1}{t^2}\bigl[f(x^* + tp) - f(x^*)\bigr] = \tfrac{1}{2} p^{\mathrm{T}} \nabla^2 f(x^*) p$$

for all nonzero $p \in \mathbb{R}^n$. (This is a restatement of (2.3).) The assumption that $\nabla^2 f(x^*)$ is positive-definite means that $p^{\mathrm{T}} \nabla^2 f(x^*) p > 0$ for all nonzero p, and it follows that $f(x^* + tp) - f(x^*) > 0$ for all $p \neq 0$ and all sufficiently small nonzero t. Because p and t are arbitrary, we conclude that there exists a neighborhood of x^* with the property that $f(x^*)$ is strictly less than f at every point in this neighborhood. Hence x^* satisfies Definition 2.1.3 of a strict local minimizer.

The next step is to show by contradiction that x^* is an isolated minimizer, i.e., that x^* is an isolated stationary point. This implies that there is a neighborhood of x^* in which there are no other stationary points of f. As x^* is an interior point where $\nabla^2 f$ exists, the gradient ∇f exists in a neighborhood of x^* and is continuous at x^*. Suppose that x^* is not an isolated stationary point, so that every neighborhood of x^* contains at least one other stationary point, i.e., a point where ∇f necessarily vanishes. Consequently there is an infinite sequence of stationary points, say $\{\bar{x}_k\}$, converging to x^*, i.e.,

$$\lim_{k\to\infty} \bar{x}_k = x^*, \ \text{ with } \nabla f(\bar{x}_k) = 0.$$

If $\|\cdot\|$ denotes the vector two-norm, the existence of $\nabla^2 f$ at x^* implies that

$$\lim_{\|p\|\to 0} \frac{1}{\|p\|} \|\nabla f(x^* + p) - \nabla f(x^*) - \nabla^2 f(x^*)p\| = 0.$$

As $\bar{x}_k - x^* \to 0$, this relation (combined with the relations $\nabla f(x^*) = 0$ and $\nabla f(\bar{x}_k) = 0$) holds with the vector p_k such that $p_k = \bar{x}_k - x^*$. Letting $u_k = p_k/\|p_k\|$ (so that $\|u_k\| = 1$), we have

$$\lim_{k\to\infty} \|\nabla^2 f(x^*)u_k\| = 0. \tag{2.5}$$

Because $\nabla^2 f(x^*)$ is positive definite, it is necessarily nonsingular, so that

$$\|\nabla^2 f(x^*)u\| \geq \sigma > 0$$

for any vector u of unit-norm, where σ is the smallest singular value of $\nabla^2 f(x^*)$. This contradicts (2.5) and shows that there is a neighborhood of x^* within which x^* is the only stationary point. As we have already shown that x^* is a minimizer, the absence of other stationary points in a neighborhood of x^* implies that x^* must be an isolated minimizer. □

The above optimality conditions can be helpful not just in recognizing a solution, but also in designing optimization algorithms. Based on the definition, if an interior point of the domain $\mathcal{D}$, say $\bar{x}$, is not a minimizer, then every neighborhood of $\bar{x}$ must contain points where the values of f is strictly less than $f(\bar{x})$. Thus there must exist at least one path along which one can move away from a non-minimizer and strictly reduces f, and a straightforward choice for such a path is a straight line. To make it precise, a *direction of decrease* is defined as a vector p along which any sufficiently small positive move produces a strictly lower value of f.

Definition 2.1.6 (direction of decrease)**.** *Let $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$ be continuous on $\mathcal{D}$. A vector $p \in \mathbb{R}^n$ is a* direction of decrease *for f at an interior point $x \in \mathcal{D}$ if there exists a positive $\hat{\alpha}$ such that $x + \hat{\alpha}p \in \mathcal{D}$ and $f(x + \alpha p) < f(x)$ for all $\alpha \in (0, \hat{\alpha})$.*

The next result provides verifiable conditions that characterize directions of decrease in two circumstances: when f is continuously differentiable and when f has a second-order derivative.

Proposition 2.1.1 ([54])**.** *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$, assume that f is continuously differentiable on a convex set $\mathcal{D}_0 \subset \mathcal{D}$, and let x be an interior point of $\mathcal{D}_0$.*

- (**a**) *If the vector p satisfies $\nabla f(x)^{\mathrm{T}} p < 0$, then p is a direction of decrease for f at x.*
- (**b**) *If, in addition, f has a second-order derivative at x, then any $\hat{p}$ such that $\nabla f(x)^{\mathrm{T}} \hat{p} < 0$ and $\hat{p}^{\mathrm{T}} \nabla^2 f(x) \hat{p} < 0$ is a direction of decrease for f at x.*

Proof. Because x is an interior point of $\mathcal{D}$ and ∇f is continuous on $\mathrm{int}(\mathcal{D})$, the fact that $\nabla f(x)^{\mathrm{T}} p < 0$ implies existence of $\delta > 0$ such that $x + \alpha p \in \mathcal{D}$ and $\nabla f(x + \alpha p)^{\mathrm{T}} p < 0$ for all $0 \le \alpha < \delta$. By the mean-value theorem for scalar-valued functions, for every α satisfying $0 < \alpha < \delta$ there is a corresponding $\widehat{t} \in (0, 1)$ such that

$$f(x + \alpha p) - f(x) = \alpha \, \nabla f(x + \widehat{t} \alpha p)^{\mathrm{T}} p.$$

As $0 < \widehat{t} \alpha < \alpha < \delta$, it follows that $f(x + \alpha p) - f(x) < 0$, which proves part (**a**).

Before proving part (**b**), observe that any direction p for which $\nabla f(x)^{\mathrm{T}} p > 0$ cannot possibly be a direction of decrease. As we know that any p for which $\nabla f(x)^{\mathrm{T}} p < 0$ is automatically a direction of decrease, we need to consider only p satisfying $\nabla f(x)^{\mathrm{T}} p = 0$.

Suppose now that f has a second derivative at x. The second-order Taylor-series expansion implies that, for any p,

$$\lim_{\alpha \to 0} \frac{1}{\alpha^2} \big[f(x + \alpha p) - f(x) - \alpha \nabla f(x)^{\mathrm{T}} p \big] = \tfrac{1}{2} p^{\mathrm{T}} \nabla^2 f(x) p.$$

For any vector $\hat{p}$ satisfying the properties given in ((**b**)), namely that $\nabla f(x)^{\mathrm{T}} \hat{p} = 0$ and $\hat{p}^{\mathrm{T}} \nabla^2 f(x) \hat{p} < 0$, it follows immediately that

$$\lim_{\alpha \to 0} \frac{1}{\alpha^2} \big[f(x + \alpha \hat{p}) - f(x) \big] = \tfrac{1}{2} \hat{p}^{\mathrm{T}} \nabla^2 f(x) \hat{p} < 0,$$

showing that $f(x + \alpha \hat{p}) - f(x) < 0$ for all sufficiently small $|\alpha|$. Consequently, if such a vector $\hat{p}$ exists, it is a direction of decrease, which verifies part (**b**). □

Two important directions of decrease are defined as follows.

Definition 2.1.7 (Descent direction)**.** *Let $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$ be continuously differentiable at x, an interior point of $\mathcal{D}$. The vector p is a* descent direction *for f at x if $\nabla f(x)^{\mathrm{T}} p < 0$.*

Definition 2.1.8 (Direction of negative curvature). *Let $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$ have a second derivative at x, an interior point of $\mathcal{D}$. The vector p is a* direction of negative curvature *for f at x if $p^{\mathrm{T}}\nabla^2 f(x)p < 0$.*

Definition 2.1.9 (Level set). *Given $f : \mathcal{D} \subset \mathbb{R}^n \mapsto \mathbb{R}$ and a scalar γ, the* level set $\mathcal{L}(\gamma)$ *is the set of points $x \in \mathcal{D}$ for which $f(x) \le \gamma$, i.e.,*

$$\mathcal{L}(\gamma) = \left\{ x \in \mathcal{D} : f(x) \le \gamma. \right\}$$

Next two definitions characterize the rate of convergence of a sequence.

Definition 2.1.10 (Q-order convergence). *A sequence $\left\{ x_k \right\}$ is said to converge to x^* with* Q-order *at least $r/ge1$ if there exist constants $\beta_r (\ge 0)$ and $K_r (\ge 0)$ such that, for all $k \ge K_r$, it holds that*

$$\|x_{k+1} - x^*\| \le \beta_r \|x_k - x^*\|^r.$$

For $r = 2$ and $r = 3$, the convergence is said to be at least Q-quadratic and Q-cubic respectively.

Definition 2.1.11 (Q-superorder convergence). *A sequence $\left\{ x_k \right\}$ is said to converge to x^* with* Q-superorder *at least $r \ge 1$ if, for every positive β_r there exists a constant $K_{\beta_r} (\ge 0)$ such that, for all $k \ge K_{\beta_r}$, it holds that*

$$\|x_{k+1} - x^*\| \le \beta_r \|x_k - x^*\|^r.$$

For $r = 1, 2$ and $= 3$, the convergence is said to be at least Q-superlinear, Q-superquadratic and Q-supercubic respectively.

Two important classes of methods for unconstrained optimization are line-search methods and trust-region methods. For a review of trust-region methods, see, e.g., M oré [69] and Conn, Gould and Toint [17]. The rest of this section will focus on line-search methods.

A line-search method is an iterative approach that produces a sequence $\left\{ x_k \right\}$ that has the form $x_{k+1} = x_k + \alpha_k p_k$, where p_k is an n-dimensional vector usually called the search direction, and α_k is a positive scalar step chosen through a line search along p_k.

The computation of both p_k and α_k in a typical line-search method depends on local model functions. Suppose that f is twice-continuously differentiable at an interior point x_k of $\mathcal{D}$. A second-order Taylor-series expansion of $f(x)$ at x_k can be written as

$$f(x_k + d) = f(x_k) + \nabla f(x_k)^{\mathrm{T}} d + \tfrac{1}{2} d^{\mathrm{T}} \nabla^2 f(x_k) d + o(\|d\|^2).$$

When $\|d\|$ is small enough so that the last two terms in the expansion can be treated as negligible, letting $d = x - x_k$ gives the Taylor-series affine model of f near x_k:

$$\ell_k(x) \triangleq f(x_k) + \nabla f(x_k)^{\mathrm{T}}(x - x_k), \tag{2.6}$$

which has the property that $\ell_k(x_k) = f(x_k)$ and $\nabla \ell_k(x_k) = \nabla f(x_k)$. When the $o(\|d\|^2)$ term can be considered negligible, $f(x)$ can be approximated by the Taylor-series quadratic model of f near x_k:

$$q_k(x) \triangleq f(x_k) + \nabla f(x_k)^{\mathrm{T}}(x - x_k) + \tfrac{1}{2}(x - x_k)^{\mathrm{T}}\nabla^2 f(x_k)(x - x_k),$$

which, like the affine model, satisfies $q_k(x_k) = f(x_k)$ and $\nabla q_k(x_k) = \nabla f(x_k)$ with the further property that $\nabla^2 q_k(x_k) = \nabla^2 f(x_k)$. A more general quadratic model that is commonly used is of the form

$$q_k(x) = f(x_k) + \nabla f(x_k)^{\mathrm{T}}(x - x_k) + \tfrac{1}{2}(x - x_k)^{\mathrm{T}} H_k(x - x_k), \tag{2.7}$$

where H_k is some symmetric approximation to $\nabla^2 f(x_k)$.

2.1.2 Choices of the step length

To choose an appropriate step length α_k for a line-search method, in often cases certain conditions are imposed on α_k to ensure convergence of the method.

Associated with the k-th iteration of a conventional line-search method for unconstrained optimization is a scalar-valued function $m_k(x)$ that represents a *local line-search model* of f. The step length α_k is then chosen to give a decrease in f that is at least as good as a fixed fraction of the decrease in the local model, i.e., α_k must satisfy

$$f(x_k) - f(x_k + \alpha_k p_k \geq \eta_{\scriptscriptstyle A}(m_k(x_k) - m_k(x_k + \alpha_k p_k)) > 0, \tag{2.8}$$

where $\eta_{\scriptscriptstyle A}$ is a fixed parameter such that $0 < \eta_{\scriptscriptstyle A} < 1$. The line-search model m_k must satisfy two conditions. First, it must hold that $m_k(x_k) - m_k(x_k + \alpha p_k) > 0$ for all α sufficiently small, i.e., the model must predict a decrease in the objective for small α. Second, m_k must be such that

$$\lim_{\alpha \to 0^+} \frac{f(x_k) - f(x_k + \alpha p_k)}{m_k(x_k) - m_k(x_k + \alpha p_k)} = 1. \tag{2.9}$$

These two conditions ensure that (2.8) is satisfied if α_k is sufficiently small Typical line-search models are the affine model (2.6) and the quadratic model (2.7) based on the first- and second-order Taylor-series approximations of f.

The affine model (2.6) gives the predicted reduction in f as

$$\ell_k(x_k) - \ell_k(x_k + \alpha p_k) = f(x_k) - (f(x_k) + \alpha \nabla f(x_k)^{\mathrm{T}} p_k) = -\alpha \nabla f(x_k)^{\mathrm{T}} p_k.$$

This prediction is strictly positive provided that p_k is a descent direction. In this case, the sufficient decrease condition (2.8) may be simplified as

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \alpha_k \eta_A \nabla f(x_k)^{\mathrm{T}} p_k, \tag{2.10}$$

which is known as the *Armijo condition* (see, e.g., Armijo [3], Ortega and Rheinboldt [75]). A line search based on the Armijo condition is known as the *Armijo line search.* A step α that satisfies the Armijo condition is called an *Armijo step.* The limit given in (2.9) implies that the Armijo condition holds if α_k is sufficiently small. Most Armijo line searches are implemented as a simple backtracking procedure in which an initial step is reduced by a constant factor until the Armijo condition (2.10) is satisfied.Thus an Armijo step has the form $\alpha_k = \gamma \sigma^{t_k}$, with $\gamma > 0$ an initial step, σ a constant factor for backtracking that satisfies $0 < \sigma < 1$, and t_k is the smallest positive integer such that α_k satisfies the Armijo condition.

A sufficient decrease criterion can be based on the minimization of $f(x_k + \alpha_k p_k)$ by recalling that, if α is a minimizer of $f(x_k + \alpha p_k)$, the directional directive $\nabla f(x_k + \alpha p_k)^{\mathrm{T}} p_k$ must be zero. As an approximation to the exact line search that seeks a minimizer of $f(x_k + \alpha p_k)$, we can require that the magnitude of $\nabla f(x_k + \alpha p_k)^{\mathrm{T}} p_k$ be sufficiently reduced compared to $\nabla f(x_k)^{\mathrm{T}} p_k$. The *first Wolfe condition* on α_k can be written as

$$|\nabla f(x_k + \alpha_k p_k)^{\mathrm{T}} p_k| \leq \eta_W |\nabla f(x_k)^{\mathrm{T}} p_k|, \tag{2.11}$$

where η_W is a preassigned parameter such that $0 \leq \eta_W < 1$ (see, e.g., Wolfe [86], Moré and Thuente [71], and Gill, Murray, Saunders and Wright [51]). The above condition does not involve the value of f, which implies that the Armijo condition (2.10) is also needed to ensure that α_k gives a sufficient decrease in f. Together, the two conditions (2.10) and (2.11) are called the *strong Wolfe conditions.* A line search based on the strong Wolfe condition is known as the *Wolfe line search.* A step α that satisfies the strong Wolfe conditions is called a *Wolfe step.*

The Wolfe conditions allow η_W to be chosen to vary the accuracy of the step. If η_A is fixed at a value close to zero (e.g., 10^{-4}), then a value of η_W close to η_A gives a "tighter" or more accurate step with respect to closeness to a critical point of $\nabla f(x_k + \alpha p_k)^{\mathrm{T}} p_k$. A value of η_W close to 1 results in a “looser” or more approximate step. A Wolfe line search is able to exploit sophisticated safeguarded polynomial interpolation techniques to provide methods that are more reliable and efficient than those based on backtracking (see, e.g., Hager [60] and Morè and Thuente [71]). The next result shows that under relatively mild conditions on f, an interval of positive steps satisfying the strong Wolfe conditions exists as long as $\eta_W \geq \eta_A$.

Proposition 2.1.2 (Existence of a Wolfe step [54])**.** *Let f be a scalar-valued twice-continuously differentiable function defined on an open convex set $\mathcal{D} \subset \mathbb{R}^n$. Consider a line-search algorithm with initial point x_0, such that the level set $\mathcal{L}(f(x_0))$ is bounded, and assume that p_k is a descent direction for all $k \geq 0$. If η_A and η_W are fixed scalars such that $0 < \eta_A \leq \eta_W < 1$, then at every iteration k, there exists an $\alpha_L^{(k)} > 0$ and an interval $(\alpha_L^{(k)}, \alpha_U^{(k)})$ such that every $\alpha \in (\alpha_L^{(k)}, \alpha_U^{(k)})$ is a Wolfe step.*

Proof. To simplify the notation, define the univariate function $\phi(\alpha) = f(x_k + \alpha p_k)$, with $\phi'(\alpha) = \nabla f(x_k + \alpha p_k)^{\mathrm{T}} p_k$. The strong Wolfe conditions are then

$$|\phi'(\alpha)| \leq \eta_W |\phi'(0)| \quad \text{and} \quad \phi(\alpha) \leq \phi(0) + \alpha \eta_A \phi'(0). \tag{2.12}$$

A local affine model of the one-dimensional line search function $\phi(\alpha)$ is a straight line emanating from $\alpha = 0$ with negative slope $\phi'(0)$, with the form $\phi(0) + \alpha\, \phi'(0)$.

Consider the function $\omega(\alpha) = \phi(\alpha) - \phi(0) - \alpha \eta_A \phi'(0)$. This function has the property that $\omega(\alpha) \leq 0$ for all α that satisfy the Armijo condition (2.10). The proof is in two parts. First, we show the existence of a positive step ξ such that $\omega'(\xi) = 0$ and $\omega(\xi) < 0$. Next we show that ξ satisfies the Wolfe conditions (2.12). First, we show that there exists a positive scalar σ such that $\omega(\alpha) < 0$ for all $\alpha \in (0, \sigma)$. Differentiating $\omega(\alpha)$ with respect to α gives $\omega'(\alpha) = \phi'(\alpha) - \eta_A \phi'(0)$, so that

$$\omega'(0) = \phi'(0) - \eta_A \phi'(0) = (1 - \eta_A)\phi'(0) < 0,$$

where we have used the assumptions that $\phi'(0) < 0$ and $\eta_A < 1$. The nonzero derivative theorem then implies that there exists a scalar σ ($\sigma > 0$) such that $\omega(\alpha) < 0$ for all $\alpha \in (0, \sigma)$. Hence, there exists a $\sigma_1 \in (0, \sigma)$ such that $\omega(\sigma_1) < 0$.

From the compactness of the level set, $\phi(\alpha)$ is bounded below by some constant ϕ_{low}, i.e., $\phi(\alpha) \geq \phi_{\text{low}}$ for all $\alpha \in [0, \infty)$. As $\phi(0) + \alpha \eta_A \phi'(0) \to -\infty$ as $\alpha \to +\infty$, there must exist a positive σ_2 such that $\phi(0) + \sigma_2 \eta_A \phi'(0) = \phi_{\text{low}}$, and we have

$$\omega(\sigma_2) = \phi(\sigma_2) - \phi(0) - \sigma_2 \eta_A \phi'(0) \geq \phi_{\text{low}} - \phi(0) - \sigma_2 \eta_A \phi'(0) = 0.$$

Given scalars σ_1 and σ_2 ($0 \le \sigma_1 < \sigma_2$) such that $\omega(\sigma_1) < 0$ and $\omega(\sigma_2) > 0$, the intermediate-value theorem states that there must exist at least one positive α such that $\omega(\alpha) = 0$. Let β denote the *smallest* positive root of $\omega(\alpha) = 0$. As $\omega(0) = 0$, $\omega(\beta) = 0$, and $\omega(\alpha) < 0$ for all $\alpha \in (0, \beta)$, the mean-value theorem implies the existence of an $\xi \in (0, \beta)$ such that $\omega'(\xi) = 0$ and $\omega(\xi) < 0$, or, equivalently,

$$\phi'(\xi) = \eta_{A}\phi'(0) \ \text{ and } \ \phi(\xi) < \phi(0) + \xi\eta_{A}\phi'(0).$$

As $\phi'(0) < 0$, we must have $\phi'(\xi) < 0$. Moreover, the inequality $\eta_{A} \le \eta_{W}$ implies that

$$\phi'(\xi) \ge \eta_{A}\phi'(0) \ge \eta_{W}\phi'(0).$$

Putting all these inequalities together, we have

$$\eta_{W}\phi'(0) \le \phi'(\xi) \le 0 \le -\eta_{W}\phi'(0),$$

or equivalently, $|\phi'(\xi)| \le \eta_{W}|\phi'(0)|$, which implies that ξ satisfies $|\phi'(\alpha)| \le \eta_{W}|\phi'(0)|$.

We have shown that ξ satisfies both the Wolfe conditions, so the set of points satisfying the Wolfe conditions (2.11) and (2.10) is non-empty. □

The next theorem shows the convergence of an Armijo line-search method.

Theorem 2.1.4 (Armijo line search [54]). *Let f be a scalar-valued twice-continuously differentiable function defined on an open convex set $\mathcal{D} \subset \mathbb{R}^n$. Assume that $x_0 \in \Omega$ is chosen such that the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded. Assume that $\{\, x_k \,\}$ is defined by $x_{k+1} = x_k + \alpha_k p_k$, where p_k is a descent direction such that $\|p_k\| \le \theta$ for some constant θ independent of k, and α_k is an Armijo step Then*

$$\lim_{k\to\infty} \big|\nabla f(x_k)^{\mathrm{T}} p_k\big| = 0.$$

Proof. Observe that the Armijo condition implies that

$$f(x_k) - f(x_{k+1}) \ge -\eta_{A}\alpha_k \nabla f(x_k)^{\mathrm{T}} p_k = \eta_{A}\alpha_k |\nabla f(x_k)^{\mathrm{T}} p_k|,$$

because $|\nabla f(x_k)^{\mathrm{T}} p_k| = -\nabla f(x_k)^{\mathrm{T}} p_k$. This relation implies that $\{x_k\}$ is well defined and remains in $\mathcal{L}(f(x_0))$. The assumption that f is bounded below on $\mathcal{L}(f(x_0))$ implies that $\{f(x_k)\}$ is a bounded, strictly decreasing sequence, and hence converges.

The rest of the proof is by contradiction. Assume that $|\nabla f(x_k)^{\mathrm{T}} p_k|$ does not converge to zero. This implies that there must exist a positive ϵ sufficiently small such that $|\nabla f(x_k)^{\mathrm{T}} p_k| \ge \epsilon$

infinitely often. Let ϵ be such a number and let $\mathcal{G}$ denote the infinite subsequence $\mathcal{G} = \big\{ k : |\nabla f(x_k)^{\mathrm{T}} p_k| \geq \epsilon \big\}$. The step length for an Armijo line search is of the form $\alpha_k = \gamma_{\scriptscriptstyle C}^{j_k}$, where j_k is the smallest nonnegative integer such that

$$f(x_k) - f(x_k + \alpha_k p_k) \geq -\alpha_k \eta_{\scriptscriptstyle A} \nabla f(x_k)^{\mathrm{T}} p_k.$$

As $f(x_k) - f(x_k + \alpha_k p_k) \to 0$ and $|\nabla f(x_k)^{\mathrm{T}} p_k| \geq \epsilon$, we must have $\alpha_k \to 0$. The assumption that the sequence of directions $\{p_k\}$ is uniformly bounded implies that the sequence of vectors $\{\alpha_k p_k\}$ must converge to zero.

Let $\bar{\mathcal{G}}$ denote the indices of those iterations at which a reduction in the initial step length was necessary, i.e., $\bar{\mathcal{G}} = \big\{ k : j_k > 0,\ k \in \mathcal{G} \big\}$. As $\{\alpha_k\}$ converges to zero for $k \in \mathcal{G}$, $\bar{\mathcal{G}}$ must be an infinite set. For every $k \in \bar{\mathcal{G}}$, define the step $\sigma_k = \alpha_k / \gamma_{\scriptscriptstyle C}$, which is the "last" step to fail the sufficient reduction test. Then, by definition,

$$f(x_k + \sigma_k p_k) > f(x_k) + \sigma_k \eta_{\scriptscriptstyle A} \nabla f(x_k)^{\mathrm{T}} p_k, \quad k \in \bar{\mathcal{G}}.$$

Adding $-\sigma_k \nabla f(x_k)^{\mathrm{T}} p_k$ to both sides of this inequality and rearranging gives

$$\begin{aligned} f(x_k + \sigma_k p_k) - f(x_k) - \sigma_k \nabla f(x_k)^{\mathrm{T}} p_k &> -\sigma_k (1 - \eta_{\scriptscriptstyle A}) \nabla f(x_k)^{\mathrm{T}} p_k \\ &> \sigma_k (1 - \eta_{\scriptscriptstyle A}) \epsilon, \quad k \in \bar{\mathcal{G}}. \end{aligned} \tag{2.13}$$

The Taylor-series expansion of $f(x_k + \sigma_k p_k)$ gives

$$f(x_k + \sigma_k p_k) - f(x_k) - \sigma_k \nabla f(x_k)^{\mathrm{T}} p_k = \sigma_k \int_0^1 \big(\nabla f(x_k + t\sigma_k p_k) - \nabla f(x_k) \big)^{\mathrm{T}} p_k \, dt. \tag{2.14}$$

If $\|\cdot\|_{\scriptscriptstyle D}$ denotes the norm dual to $\|\cdot\|$, i.e., $\|x\|_{\scriptscriptstyle D} = \max_{v \neq 0} |x^{\mathrm{T}} v| / \|v\|$, then the generalized Cauchy-Schwartz inequality gives

$$\big| \big(\nabla f(x_k + t\sigma_k p_k) - \nabla f(x_k) \big)^{\mathrm{T}} p_k \big| \leq \| \nabla f(x_k + t\sigma_k p_k) - \nabla f(x_k) \|_{\scriptscriptstyle D} \|p_k\|.$$

If this inequality is substituted in (2.14) and standard norm inequalities are applied, the inequality (2.13) implies that

$$(1 - \eta_{\scriptscriptstyle A}) \epsilon < \int_0^1 \big(\nabla f(x_k + t\sigma_k p_k) - \nabla f(x_k) \big)^{\mathrm{T}} p_k \, dt \leq \max_{0 \leq t \leq 1} \| \nabla f(x_k + t\sigma_k p_k) - \nabla f(x_k) \|_{\scriptscriptstyle D} \, \|p_k\|,$$

for every $k \in \bar{\mathcal{G}}$. The continuity of ∇f implies that there exists some $\theta \in (0, \alpha_k/\gamma_C)$ such that

$$\max_{0 \le t \le 1} \|\nabla f(x_k + t\sigma_k p_k) - \nabla f(x_k)\|_D = \|\nabla f(x_k + \theta p_k) - \nabla f(x_k)\|_D.$$

It follows that

$$(1 - \eta_A)\epsilon < \|\nabla f(x_k + \theta p_k) - \nabla f(x_k)\|_D \, \|p_k\|. \tag{2.15}$$

However, $\alpha_k p_k \to 0$ on the , in which case it must hold that $\theta p_k \to 0$, and the continuity of $\nabla f(x)$ gives

$$\|\nabla f(x_k + \theta p_k) - \nabla f(x_k)\|_D \to 0.$$

The assumption on the boundedness of $\{p_k\}$ implies that the right-hand side of (2.15) converges to zero, which gives the required contradiction. □

The convergence of a Wolfe line-search method is stated below.

Theorem 2.1.5 (Wolfe line search [54])**.** *Let f be a scalar-valued twice-continuously differentiable function defined on an open convex set $\mathcal{D} \subset \mathbb{R}^n$. Assume that $x_0 \in \mathit{\Omega}$ is chosen such that the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded. Assume that $\{\, x_k \,\}$ is defined by $x_{k+1} = x_k + \alpha_k p_k$, where p_k is a descent direction such that $\|p_k\| \le \theta$ for some constant θ independent of k, and α_k is a step that satisfies the strong Wolfe conditions. Then*

$$\lim_{k \to \infty} \big|\nabla f(x_k)^{\mathrm{T}} p_k\big| = 0.$$

Proof. The first Wolfe condition is equivalent to the Armijo backtracking condition, and the arguments of Theorem 2.1.4 may be used to show that $\{f(x_k)\}$ is a convergent sequence.

As in Theorem 2.1.4, the main part of the proof is by contradiction. If $|\nabla f(x_k)^{\mathrm{T}} p_k|$ does not converge to zero, then there must be a positive ϵ sufficiently small such that $|\nabla f(x_k)^{\mathrm{T}} p_k| \ge \epsilon$ infinitely often. If ϵ is such a value, let $\mathcal{G} = \{\, k : |\nabla f(x_k)^{\mathrm{T}} p_k| \ge \epsilon \,\}$. With this assumption, the first Wolfe condition gives

$$f(x_k) - f(x_{k+1}) \ge \eta_A \alpha_k \epsilon, \quad \text{for all } k \in \mathcal{G}. \tag{2.16}$$

As $\{f(x_k)\}$ is a convergent sequence, the left-hand side of (2.16) converges to zero for $k \in \mathcal{G}$. The definitions of η_A and ϵ imply that $\alpha_k \to 0$ for $k \in \mathcal{G}$, and the uniform boundedness of the sequence $\{p_k\}$ gives $\alpha_k p_k \to 0$ for $k \in \mathcal{G}$.

The second Wolfe condition may be rearranged to be of the form

$$|\nabla f(x_k+\alpha_k p_k)^{\mathrm{T}} p_k| \le \eta_W |\nabla f(x_k)^{\mathrm{T}} p_k|, \text{ so that } -|\nabla f(x_k+\alpha_k p_k)^{\mathrm{T}} p_k| \ge \eta_W |\nabla f(x_k)^{\mathrm{T}} p_k|.$$

It follows that

$$\begin{aligned}|\nabla f(x_k)^{\mathrm{T}} p_k| + \nabla f(x_k+\alpha_k p_k)^{\mathrm{T}} p_k &\ge |\nabla f(x_k)^{\mathrm{T}} p_k| - |\nabla f(x_k+\alpha_k p_k)^{\mathrm{T}} p_k| \\ &\ge (1-\eta_W)|\nabla f(x_k)^{\mathrm{T}} p_k|.\end{aligned}$$

As $|\nabla f(x_k)^{\mathrm{T}} p_k| = -\nabla f(x_k)^{\mathrm{T}} p_k$, we have

$$\nabla f(x_k+\alpha_k p_k)^{\mathrm{T}} p_k - \nabla f(x_k)^{\mathrm{T}} p_k \ge (1-\eta_W)|\nabla f(x_k)^{\mathrm{T}} p_k| \ge (1-\eta_W)\epsilon, \qquad k \in \mathcal{G}. \tag{2.17}$$

Using standard norm inequalities, we obtain

$$\begin{aligned}\nabla f(x_k+\alpha_k p_k)^{\mathrm{T}} p_k - \nabla f(x_k)^{\mathrm{T}} p_k &= \big(\nabla f(x_k+\alpha_k p_k) - \nabla f(x_k)\big)^{\mathrm{T}} p_k \\ &\le \|\nabla f(x_k+\alpha_k p_k) - \nabla f(x_k)\|_D \, \|p_k\|, \end{aligned} \tag{2.18}$$

where $\|\cdot\|_D$ denotes the norm dual to $\|\cdot\|$. Inequalities (2.17) and (2.18) imply that

$$\|\nabla f(x_k+\alpha_k p_k) - \nabla f(x_k)\|_D \, \|p_k\| \ge (1-\eta_W)\epsilon > 0, \qquad k \in \mathcal{G}. \tag{2.19}$$

Because $1-\eta_W$ and ϵ are bounded away from zero, this inequality implies that the vector difference inside the norm on the left is bounded away from zero. But we know that $\alpha_k p_k \to 0$, for all $k \in \mathcal{G}$, and the continuity of $\nabla f(x)$ and the boundedness of $\{p_k\}$ imply that the left-hand side of (2.19) must converge to zero. This gives the desired contradiction, and shows that $|\nabla f(x_k)^{\mathrm{T}} p_k| \to 0$. □

2.1.3 Implementing the Wolfe line search

A typical implementation of the Wolfe line search may be viewed as a two-stage process. The first stage involves the determination of an interval containing a Wolfe step, if one exists. The second stage locates a Wolfe step in this interval using safeguarded polynomial interpolation. If the first stage fails, then the objective function is necessarily unbounded below. The key principle that drives the first stage is that certain conditions may be formulated that determine if an interval contains a Wolfe step. Much of the discussion in this section is based on the work of Moré and

Sorensen [70], Morè and Thuente [71]. More information may be found in Wolfe [87]. The schematic description of the line-search algorithm given in Algorithm 1 below follows that of Nocedal and Wright [74]. In order to simplify the notation we omit the suffix k and consider the univariate function $\phi(\alpha) = f(x+\alpha p)$ for fixed vectors x and p. With this notation the Wolfe conditions (2.10) and (2.11) may be written in the form

$$\phi(\alpha) \le \phi(0) + \alpha\eta_{\scriptscriptstyle A}\phi'(0), \quad \text{and} \quad |\phi'(\alpha)| \le \eta_{\scriptscriptstyle W}|\phi'(0)|.$$

Much of the theory associated with a Wolfe line search is based on the properties of the auxiliary function

$$\omega(\alpha) = \phi(\alpha) - \bigl(\phi(0) + \alpha\eta_{\scriptscriptstyle A}\phi'(0)\bigr), \quad \text{with} \quad \omega'(\alpha) = \phi'(\alpha) - \eta_{\scriptscriptstyle A}\phi'(0).$$

Moré and Sorensen [70] show that a minimizer of this function at which ω is negative satisfies the Wolfe conditions. An example of a function ϕ and its associated auxiliary function ω are depicted in Figure 2.1. The first stage of a Wolfe line search is motivated by the following proposition.

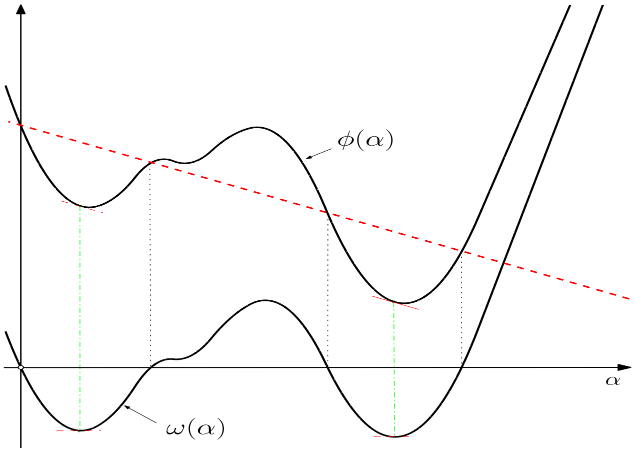

Figure 2.1: The graph depicts $\phi(\alpha) = f(x+\alpha p)$ as a function of positive α, with the shifted function $\omega(\alpha) = \phi(\alpha) - \bigl(\phi(0) + \alpha\eta_{\scriptscriptstyle A}\phi'(0)\bigr)$ superimposed. The dashed line represents the affine function $\phi(0) + \alpha\eta_{\scriptscriptstyle A}\phi'(0)$.

Proposition 2.1.3. *Let* $\{\,\alpha_i\,\}_{i=0}^{\infty}$ *be a strictly monotonically increasing sequence with* $\alpha_0 = 0$. *Let* ϕ *and* ω *be continuously differentiable univariate functions such that* $\phi'(0) < 0$ *and* $\omega(\alpha) = \phi(\alpha) - \bigl(\phi(0) + \alpha\eta_{\scriptscriptstyle A}\phi'(0)\bigr)$ *with* $0 < \eta_{\scriptscriptstyle A} < 1$. *If there exists a least bounded index* j *such that at least one of the following conditions is true:*

(**a**) α_j *is a Wolfe step;*

(**b**) $\omega(\alpha_j) \geq \omega(\alpha_{j-1})$*; or*

(**c**) $\omega'(\alpha_j) \geq 0$*,*

then there exists a Wolfe step $\alpha^* \in [\alpha_{j-1}, \alpha_j]$*. Collectively,* (**a**)–(**c**) *are called the* stage-one conditions.

Proof. Observe that α_{j-1} must satisfy none of the conditions (**a**)–(**c**), otherwise j would not be the least index. This implies that $\omega(\alpha_{j-1}) < \omega(\alpha_{j-2}) < \cdots < \omega(\alpha_0) = 0$ from (**b**), and $\omega'(\alpha_{j-1}) < 0$ from (**c**).

Case 1. If (**a**) is true, the proposition is true trivially.

Case 2. If (**b**) is true, let $\bar{\alpha} = \sup\big\{\, \alpha \in [\alpha_{j-1}, \alpha_j] : \omega(\beta) \leq 0 \;\text{ for all }\; \beta \in [\alpha_{j-1}, \alpha] \,\big\}$. If $\bar{\alpha} = \alpha_j$, then $\omega(\bar{\alpha}) = \omega(\alpha_j) \geq \omega(\alpha_{j-1})$; if $\bar{\alpha} < \alpha_j$, then by the continuity of ω, $\omega(\bar{\alpha}) = 0 > \omega(\alpha_{j-1})$. From the mean-value theorem there must exist an $\widehat{\alpha} \in (\alpha_{j-1}, \bar{\alpha})$ such that $\omega'(\widehat{\alpha}) = \big(\omega(\bar{\alpha}) - \omega(\alpha_{j-1})\big)/(\bar{\alpha} - \alpha_{j-1}) > 0$. The function $\omega(\alpha)$ is continuously differentiable with $\omega'(\alpha_{j-1}) < 0$ and $\omega'(\widehat{\alpha}) > 0$. The intermediate-value theorem then implies that there must exist a step $\alpha^* \in [\alpha_{j-1}, \widehat{\alpha}]$ such that $\omega'(\alpha^*) = 0$. As $\omega(\alpha^*) \leq 0$, α^* is a Wolfe step.

Case 3. Finally, consider the case where (**c**) is true. If $\omega(\alpha) < 0$ for all $[\alpha_{j-1}, \alpha_j]$, then, as $\omega'(\alpha_{j-1}) < 0$ and $\omega'(\alpha_j) \geq 0$, the continuity of ω' and the intermediate-value theorem imply that there exists a step $\alpha^* \in [\alpha_{j-1}, \alpha_j]$ such that $\omega'(\alpha^*) = 0$. As $\omega(\alpha^*) < 0$, α^* is a Wolfe step. Otherwise, if there exists some $\alpha \in [\alpha_{j-1}, \alpha_j]$ such that $\omega(\alpha) \geq 0$, let $\bar{\alpha} = \sup\big\{\, \alpha \in [\alpha_{j-1}, \alpha_j] : \omega(\beta) \leq 0 \text{ for all } \beta \in [\alpha_{j-1}, \alpha] \,\big\}$. The continuity of ω implies that $\omega(\bar{\alpha}) = 0$. The same argument used in Case 2 may be used to show that there must exist an $\widehat{\alpha} \in (\alpha_{j-1}, \bar{\alpha})$ such that $\omega'(\widehat{\alpha}) > 0$ and an $\alpha^* \in [\alpha_{j-1}, \widehat{\alpha}]$ such that $\omega'(\alpha^*) = 0$ with $\omega(\alpha^*) \leq 0$. □

Note that the converse result is not true, e.g., there may be a Wolfe step in the interval $[0, \alpha_1]$ even though none of the stage-one conditions are satisfied for $j = 1$. The behavior of $\omega(\alpha)$ is unknown at any $\alpha \in (0, \alpha_1)$.

If the first step α_1 is not a Wolfe step, successively larger steps are computed until either one of the stage-one conditions is satisfied or j is such that $\alpha_j = \alpha_{\max}$. In practice, $\alpha_{\max}$ is an upper bound imposed on the step and the search is terminated if the bound is exceeded during the stage-one iterations. If a given α_j does not satisfy the stage-one conditions then $\omega(\alpha_j) < \omega(\alpha_{j-1}) < \cdots < \omega(\alpha_0) = 0$. If the algorithm reaches $\alpha_{j_{\max}} = \alpha_{\max}$ and none of the stage-one conditions have been satisfied, it terminates with $\alpha_{j_{\max}}$, which is an Armijo step with the least computed function value.

Proposition 2.1.3 implies that if one of the stage-one conditions is satisfied at iteration j, then the interval $[\alpha_{j-1}, \alpha_j]$ must contain a Wolfe step. At this point the line search terminates successfully if the stage-one condition (**a**) is satisfied, or moves on to the second stage. The computations associated with the second stage are based on the following result.

Proposition 2.1.4. *Let ϕ and ω be defined as in* Proposition 2.1.3. *Assume there exist distinct points α_{low} and α_{high} such that*

(**a**) $\omega(\alpha_{\text{low}}) \le 0$;

(**b**) $\omega(\alpha_{\text{low}}) \le \omega(\alpha_{\text{high}})$; *and*

(**c**) $\omega'(\alpha_{\text{low}})(\alpha_{\text{high}} - \alpha_{\text{low}}) < 0$.

Then there exists a Wolfe step $\alpha^* \in \mathcal{I}$, where $\mathcal{I}$ is the interval defined with endpoints α_{low} and α_{high}.

Proof. The proof is similar to that of Proposition 2.1.3, and is a special case of the proof of Proposition 3.2.3. □

The conditions (**a**)–(**c**) of Proposition 2.1.4 are referred to collectively as the *stage-two conditions*. The subscripts associated with the points α_{low} and α_{high} serve to emphasize the fact that $\omega(\alpha_{\text{low}}) \le \omega(\alpha_{\text{high}})$. It is not necessarily the case that $\alpha_{\text{low}} < \alpha_{\text{high}}$.

Algorithm 1 gives a schematic outline of a Wolfe line search. The calculations required for a Wolfe line search may be organized into two "functions" associated with the stage-one and stage-two conditions. If the first stage finds an interval that contains a Wolfe step, the first-stage function labels the endpoints α_{low} and α_{high} based on relative magnitudes of $\omega(\alpha_{j-1})$ and $\omega(\alpha_j)$, and calls the stage-two function `Stage_Two`$(\alpha_{\text{low}}, \alpha_{\text{high}})$. The second-stage function interpolates the endpoints to calculate a best-guess step, α_{new}, in the interval. The second-stage function is called recursively using α_{new} and an existing endpoint, labeling them so that the stage-two conditions hold for each call. This is repeated until α_{new} is a Wolfe step. In practice, it rarely takes more than one or two interpolations to find a Wolfe step. It must be emphasized that in practice, the stage-two calculations are not implemented as a recursive procedure. The recursive structure depicted in Algorithm 1 is illustrative and reflects the fact that the intervals defined by α_{low} and α_{high} form a *nested sequence.* If $\mathcal{I}_0$ is the interval resulting from stage-one, the computations of stage-two generate a sequence of intervals $\{\, \mathcal{I}_j \,\}$ and a sequence of points $\{\, \alpha_{\text{low}}^{(j)} \,\}$ such that $\alpha^{(j)} \in \mathcal{I}_j$, each $\mathcal{I}_j$

contains a quasi-Wolfe step, and $\mathcal{I}_j \subset \mathcal{I}_{j-1}$. The intervals $\mathcal{I}_j$ form a nested sequence of "intervals of uncertainty".

Algorithm 1 Schematic outline of a Wolfe line search.

```
function WOLFE_LINE_SEARCH($\alpha$)
    restriction: $\alpha > 0$;
    constants: $\eta_A \in (0, \frac{1}{2})$, $\eta_W \in (\eta_A, 1)$, $\gamma_e > 1$, $\alpha_{\max} \in (0, +\infty)$;
    $\alpha \leftarrow \min\{\alpha, \alpha_{\max}\}$;   $\alpha_{\text{old}} \leftarrow 0$;
    while $\alpha$ is not a Wolfe step and $\alpha \neq \alpha_{\max}$ do
        if $\omega(\alpha) \geq \omega(\alpha_{\text{old}})$ then
            $\alpha \leftarrow$ Stage_Two($\alpha_{\text{old}}, \alpha$); break;
        else if $\omega'(\alpha) \geq 0$ then
            $\alpha \leftarrow$ Stage_Two($\alpha, \alpha_{\text{old}}$); break;
        else
            $\alpha_{\text{old}} \leftarrow \alpha$;   $\alpha \leftarrow \min\{\gamma_e \alpha, \alpha_{\max}\}$;        [Increase $\alpha$ towards $\alpha_{\max}$]
        end if
    end while
    return $\alpha$;
end function
function STAGE_TWO($\alpha_{\text{low}}, \alpha_{\text{high}}$)
    restriction: $\omega(\alpha_{\text{low}}) \leq \omega(\alpha_{\text{high}})$;
    Choose $\alpha_{\text{new}}$ in the interior of the interval defined by $\alpha_{\text{low}}$ and $\alpha_{\text{high}}$;
    if $\alpha_{\text{new}}$ is a Wolfe step then
        return $\alpha_{\text{new}}$;
    else if $\omega(\alpha_{\text{new}}) \geq \omega(\alpha_{\text{low}})$ then
        return Stage_Two($\alpha_{\text{low}}, \alpha_{\text{new}}$);
    else if $\omega'(\alpha_{\text{new}})(\alpha_{\text{high}} - \alpha_{\text{low}}) < 0$ then
        return Stage_Two($\alpha_{\text{new}}, \alpha_{\text{high}}$);
    else
        return Stage_Two($\alpha_{\text{new}}, \alpha_{\text{low}}$);
    end if
end function
```

A practical implementation of a Wolfe line search is very complex. There are many ways to interpolate to obtain a new point in the second stage. The use of finite precision imposes the need for some sort of safeguarding during interpolation and gives rise to a whole host of issues, including how to handle cases when the function or step length are changing by a value near or less than machine precision. See, e.g., Brent [9], Hager [60], Ghosh and Hager [45], and Moré and Thuente [71] for more details.

2.1.4 Choices of the search direction

The search direction p_k for a line-search method is typically derived by minimizing a local quadratic model at x_k of the form (2.7), with H_k some positive-definite approximation of the Hessian matrix $\nabla^2 f(x_k)$. Writing $p = x - x_k$ in (2.7) gives

$$q_k(p) = f(x_k) + \nabla f(x_k)^{\mathrm{T}} p + \tfrac{1}{2} p^{\mathrm{T}} H_k p,$$

The explicit formula for p_k can be obtained by setting the gradient of $q_k(p)$ to zero:

$$p_k = -H_k^{-1} \nabla f(x_k).$$

As H_k is positive-definite,

$$\nabla f(x_k)^{\mathrm{T}} p_k = -p_k^{\mathrm{T}} H_k p_k < 0,$$

which implies that p_k is a descent direction.

Simply defining H_k as the identity matrix gives the steepest-descent direction $-\nabla f(x_k)$, along which the function decreases the fastest in the sense that $p_k = -\nabla f(x_k)$ is the solution of the following minimization problem:

$$\underset{p \in \mathbb{R}^n}{\text{minimize}} \;\; \nabla f(x_k)^{\mathrm{T}} p \quad \text{subject to} \;\; \|p_k\| / \|\nabla f(x_k)\| = 1, \tag{2.20}$$

where $\nabla f(x_k)^{\mathrm{T}} p$ represents the directional derivative of f at x_k along a direction p. The precise choice of the scaling of p_k is based on the idea of having $\|p_k\| \to 0$ as the sequence of iterates approach a stationary point, i.e., as $\|\nabla f(x_k)\| \to 0$. The steepest-descent direction is often used when the second-order derivatives of f are unavailable or are too expensive to compute.

However, the steepest-descent direction can be inadequate as its definition does not include the curvature information of f, which is crucial for locating a minimizer efficiently. The well-known Newton's method is a second-order method that uses the exact Hessian matrix at each iteration k:

$$H_k = \nabla^2 f(x_k).$$

Thus, the *Newton direction* p_k is derived by solving the linear system

$$\nabla^2 f(x_k) p_k = -\nabla f(x_k).$$

To ensure that p_k is a descent direction, this approach can only be used when $\nabla^2 f(x_k)$ is positive definite at each iteration k. As this condition does not hold for general functions, one strategy is to define H_k as a positive-definite approximation to $\nabla^2 f(x_k)$, on the assumption that the "natural" choice of $H_k = \nabla^2 f(x_k)$ will be made when $\nabla^2 f(x_k)$ is sufficiently positive-definite. Such methods are often referred to as *modified Newton methods*. A modified Hessian matrix H_k can be defined as the positive-definite matrix closest to $\nabla^2 f(x_k)$ that has a condition number no greater than a preassigned value, say β. To determine the matrix H_k is to solve the following constrained matrix problem:

$$\underset{H_k \in \mathbb{R}^{n\times n}}{\text{minimize}} \left\{ \|\nabla^2 f(x_k) - H_k\|_F : \text{cond}(H_k) \le \beta, \quad H_k \text{ symmetric positive definite } \right\},$$

where $\|\cdot\|_F$ represents the Frobenius norm and $\text{cond}(H_k)$ denotes the condition number of H (i.e., the ratio of the eigenvalues of largest and smallest absolute value). The restriction on the size of the condition number of H_k ensures that H_k is sufficiently positive-definite. This problem is often solved using the symmetric indefinite factorization of the modified Cholesky factorization.

The following theorem states the local convergence properties of the Newton's method. The proof is widely available in literature, see, e.g., Nocedal and Wright [74].

Theorem 2.1.6 (Local convergence of Newton's method [54])**.** *Let f be a scalar-valued twice-continuously differentiable function defined on an open convex set $\mathcal{D} \subset \mathbb{R}^n$, and assume that $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is nonsingular for some $x^* \in \mathcal{D}$. Then*

(**a**) *there exists a neighborhood $\mathcal{B}(x^*, \delta)$ such that, for any x_0 in $\mathcal{B}$, the sequence of Newton iterates $\{ x_k \}_{k\ge 0}$ such that*

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k) \tag{2.21}$$

are well defined, remain in $\mathcal{B}$ and converge to x^* at Q-superlinear rate;

(**b**) *if, in addition, $\nabla^2 f$ satisfies a local Lipschitz condition at x^*,*

$$\|\nabla^2 f(x) - \nabla^2 f(x^*)\| \le L\|x - x^*\| \text{ for all } x \in \mathcal{B}$$

where L is a positive constant, the Newton iterates defined by (2.21) converge Q-quadratically to x^*; and

(**c**) *if, in addition, $\nabla^2 f$ satisfies a local Lipschitz condition with constant L in a neighborhood of x^*, $\|f(x_k)\|$ converges Q-quadratically to zero.* □

Theorem 2.1.4 and Theorem 2.1.5 imply that combining a descent direction and an Armijo or Wolfe step ensures $|\nabla f(x_k)^{\mathrm{T}} p_k| \to 0$. However, to obtain convergence to a stationary point, i.e., $\nabla f(x_k) \to 0$, the search direction p_k needs to have the property that $|\nabla f(x_k)^{\mathrm{T}} p_k|$ can go to zero only when $\nabla f(x_k)$ goes to zero.

Definition 2.1.12 (Direction of sufficient descent)**.** *A direction p_k is a* direction of sufficient descent *if $\|p_k\|$ is bounded and*

$$\nabla f(x_k)^{\mathrm{T}} p_k \to 0 \ \textit{ implies } \ \nabla f(x) \to 0 \ \textit{ and } \ p_k \to 0.$$

A convenient characterization of directions of sufficient descent is provided by the following lemma.

Lemma 2.1.1. *Let $\{\, H_k \,\}$ be a sequence of symmetric positve-definite matrices such that*

$$\lambda_{max}(H_k) \le M < +\infty \ \textit{ and } \ \lambda_{min}(H_k) \ge m > 0,$$

where m and M are constants, and $\lambda_{max}(H_k)$ and $\lambda_{min}(H_k)$ denotes the largest and smallest eigenvalues of H_k respectively. If the search direction p_k is computed by solving the equations $H_k p_k = -\nabla f(x_k)$, then p_k is a direction of sufficient decrease.

Proof. Because $H_k p_k = -\nabla f(x_k)$, it follows that

$$|\nabla f(x_k)^{\mathrm{T}} p_k| = |p_k^{\mathrm{T}} H_k p_k| \ge \lambda_{\min} \|p_k\|^2, \tag{2.22}$$

where $\|\cdot\|$ denotes the two-norm. As $\lambda_{\min}(H_k) > m$ for all k we have immediately that $p_k \to 0$ when $|\nabla f(x_k)^{\mathrm{T}} p_k| \to 0$.

Applying standard norm inequalities to the expression $H_k p_k = -\nabla f(x_k)$ for p_k gives

$$\|H_k\| \, \|p_k\| \ge \|\nabla f(x_k)\|, \ \text{ so that } \ \|p_k\| \ge \frac{\|\nabla f(x_k)\|}{\|H_k\|}. \tag{2.23}$$

Using the relation $\|H_k\| = \lambda_{\max}$, we may combine (2.22) and (2.23) as follows:

$$|\nabla f(x_k)^{\mathrm{T}} p_k| \ge \frac{\lambda_{\min}}{\lambda_{\max}} \|p_k\| \, \|\nabla f(x_k)\| \ge \frac{\lambda_{\min}}{\lambda_{\max}^2} \|\nabla f(x_k)\|^2. \tag{2.24}$$

As by assumption $\lambda_{\max}(H_k) \le M$, $\lambda_{\min}(H_k) \ge m$, we have $\nabla f(x_k) \to 0$ if $|\nabla f(x_k)^{\mathrm{T}} p_k| \to 0$, as required. □

Combining Newton's method with an Armijo or Wolfe line search ensures global convergence of the Newton-based line-search algorithm.

Theorem 2.1.7 (Global convergence of Newton-based line-search method [54]). *Let f be a scalar-valued twice-continuously differentiable function defined on an open convex set $\mathcal{D} \subset \mathbb{R}^n$. Assume that $\nabla^2 f(x)$ is positive definite for all $x \in \mathcal{D}$, with smallest eigenvalue uniformly bounded away from zero; i.e., there exists $\sigma > 0$ such that for all $s \in \mathbb{R}^n$ and $x \in \mathcal{D}$,*

$$s^{\mathrm{T}} \nabla^2 f(x) s \geq \sigma \|s\|^2. \tag{2.25}$$

Given $x_0 \in \mathcal{D}$ such that the level set $\mathcal{L}(f(x_0))$ is compact, consider the sequence $x_{k+1} = x_k + \alpha_k p_k$, where p_k is the Newton direction and α_k is the Armijo or Wolfe step length. The sequence $\{\, x_k \,\}$ is well-defined and lies in $\mathcal{L}(f(x_0))$. Moreover, either the algorithm finds some x_k such that $\nabla f(x_k) = 0$, or $\lim_{k\to\infty} \nabla f(x_k) = 0$.

Proof. To show that the iterates are well defined, observe that, if $x_k \in \mathcal{L}(f(x_0))$, the positive-definiteness of $\nabla^2 f(x)$ ensures that the Newton direction is always a direction of descent. Hence, Proposition 2.1.2 ensures the existence of a suitable α_k satisfying the step length conditions (2.11) and (2.10), which implies that $x_{k+1} \in \mathcal{L}(f(x_0))$.

The smallest eigenvalue of H_k is bounded below by assumption; continuity of $\nabla^2 f(x)$ and the compactness of $\mathcal{L}(f(x_0))$ together imply that $\|H_k\|$ is bounded, so that the largest eigenvalue is bounded above. The conditions of Lemma 2.1.1 therefore apply, and $p_k^{\scriptscriptstyle N}$ is a direction of sufficient descent. The result then follows immediately from Definition 2.1.12. □

The next two theorems state that the "natural" step of unity eventually satisfies the Armijo and Wolfe conditions; this property is essential to achieving quadratic rate of convergence.

Theorem 2.1.8 (Armijo sufficient decrease with a unit step [54]). *Assume that the assumptions of* Theorem 2.1.7 *hold. Moreover, assume that the scalar $\eta_{\scriptscriptstyle A}$ is chosen so that $0 < \eta_{\scriptscriptstyle A} < \frac{1}{2}$. Then there exists an index K such that for all $k \geq K$ for which $\nabla f(x_k) \neq 0$, the step $\alpha_k = 1$ satisfies the Armijo condition.*

Proof. First, observe that (2.25) and the definition $\nabla^2 f(x_k) p_k = -\nabla f(x_k)$ imply that

$$-\nabla f(x_k)^{\mathrm{T}} p_k \geq p_k^{\mathrm{T}} \nabla^2 f(x_k) p_k \geq \sigma \|p_k\|^2. \tag{2.26}$$

As we know from Theorem 2.1.7 that $\nabla f(x_k) \to 0$, (2.26) implies that $p_k \to 0$. This condition and the descent property of p_k then imply that for all sufficiently large k, $x_k + p_k \in \mathcal{L}(f(x_0))$. We henceforth consider only such values of k.

The condition to be verified is that

$$f(x_k + p_k) - f(x_k) - \eta_A \nabla f(x_k)^{\mathrm{T}} p_k \leq 0. \tag{2.27}$$

Using the identity

$$f(x+p) = f(x) + \nabla f(x)^{\mathrm{T}} p + \tfrac{1}{2} p^{\mathrm{T}} \nabla^2 f(x) p + \int_0^1 p^{\mathrm{T}} \big(\nabla^2 f(x+tp) - \nabla^2 f(x)\big) p (1-t)\, dt,$$

and the definition $\nabla^2 f(x_k) p_k = -\nabla f(x_k)$, the left-hand side of (2.27) may be written as

$$\begin{aligned} f(x_k + p_k) - f(x_k) - \eta_A \nabla f(x_k)^{\mathrm{T}} p_k &= \frac{1}{2}(1 - 2\eta_A) \nabla f(x_k)^{\mathrm{T}} p_k \\ &\quad + \int_0^1 p_k^{\mathrm{T}} \big(\nabla^2 f(x_k + tp_k) - \nabla^2 f(x_k)\big) p_k (1-t)\, dt. \end{aligned}$$

As $\nabla^2 f$ is continuous in the closed bounded region $\mathcal{L}(f(x_0))$, the quantity ω_k such that

$$\omega_k = \max_{0 \leq t \leq 1} \|\nabla^2 f(x_k + tp_k) - \nabla^2 f(x_k)\|$$

is bounded. Applying this definition of ω_k and (2.26), gives

$$\begin{aligned} f(x_k + p_k) - f(x_k) - \eta_A \nabla f(x_k)^{\mathrm{T}} p_k &\leq \frac{1}{2}(1 - 2\eta_A) \nabla f(x_k)^{\mathrm{T}} p_k + \frac{1}{2} \omega_k \|p_k\|^2 \\ &\leq \frac{1}{2}(1 - 2\eta_A - \frac{\omega_k}{\sigma}) \nabla f(x_k)^{\mathrm{T}} p_k. \end{aligned} \tag{2.28}$$

As $p_k \to 0$, there must exist an index K such that for all $k \geq K$, ω_k will be small enough to satisfy $\omega_k < \sigma(1 - 2\eta_A)$. When ω_k satisfies this inequality, the right-hand side of (2.28) will be negative for ω sufficiently small (because $\eta_A < \frac{1}{2}$ and $\nabla f(x_k)^{\mathrm{T}} p_k < 0$), and hence (2.27) is satisfied for sufficiently large k. □

Theorem 2.1.9 (Wolfe sufficient decrease with a unit step [54]). *Under the assumptions of* Theorem 2.1.7, *there exists an index K such that for all $k \geq K$ for which $\nabla f(x_k) \neq 0$, the step $\alpha_k = 1$ satisfies the strong Wolfe conditions* (2.11) *and* (2.10).

Proof. The condition to be verified is that $|\nabla f(x_k+p_k)^{\mathrm T}p_k| \le \eta_W|\nabla f(x_k)^{\mathrm T}p_k|$. Consider the identity

$$\nabla f(x_k+p_k) - \nabla f(x_k) - \nabla^2 f(x_k)p_k = \int_0^1 \left(\nabla^2 f(x_k+tp_k) - \nabla^2 f(x_k)\right) p_k\, dt.$$

Multiplying by $p_k^{\mathrm T}$ and re-arranging, we obtain

$$\nabla f(x_k+p_k)^{\mathrm T}p_k = \nabla f(x_k)^{\mathrm T}p_k + p_k^{\mathrm T}\nabla^2 f(x_k)p_k + \int_0^1 p_k^{\mathrm T}\left(\nabla^2 f(x+tp_k) - \nabla^2 f(x_k)\right) p_k\, dt.$$

Substituting $\nabla^2 f(x_k)p_k = -\nabla f(x_k)$ in this equation and applying norm inequalities gives

$$\begin{aligned} \left|\nabla f(x_k+p_k)^{\mathrm T}p_k\right| &\le \left|\int_0^1 p_k^{\mathrm T}\left(\nabla^2 f(x+tp_k) - \nabla^2 f(x_k)\right) p_k\, dt\right| \\ &\le \|p_k\|^2 \max_{0\le t\le 1} \|\nabla^2 f(x_k+tp_k) - \nabla^2 f(x_k)\|. \end{aligned} \tag{2.29}$$

Substituting (2.26) and the definition of ω_k in (2.29), we obtain the inequality

$$\left|\nabla f(x_k+p_k)^{\mathrm T}p_k\right| \le \frac{\omega_k}{\sigma}\left|\nabla f(x_k)^{\mathrm T}p_k\right|. \tag{2.30}$$

The continuity of $\nabla^2 f$ and the fact that $p_k \to 0$, implies that there must exist an index K such that $\omega_k/\sigma < \eta_W$ for all $k \ge K$. Using this inequality in (2.30) gives that $|\nabla f(x_k+p_k)^{\mathrm T}p_k| \le \eta_W|\nabla f(x_k)^{\mathrm T}p_k|$ is satisfied by x_k+p_k for all sufficiently large k.

Again, the result that $\|p_k\| \to 0$ implies the existence of an index K such that for $k \ge K$, ω_k will be small enough to satisfy $\omega_k < \sigma\min(1-2\eta_A, \eta_W)$. When ω_k satisfies this inequality, the right-hand side of (2.28) will be negative for ω sufficiently small (because $\eta_A < \frac{1}{2}$ and $\nabla f(x_k)^{\mathrm T}p_k < 0$), and hence the Armijo condition (2.27) is satisfied for sufficiently large k.

Thus, $\alpha_k = 1$ satisfies both Wolfe conditions (2.10) and (2.11) for all sufficiently large k. □

The fast local convergence of Newton-based methods derives from the curvature information provided by the Hessian matrix at each iteration. Unfortunately, the required exact Hessian may be expensive or impossible to obtain for many functions encountered in practice. Accordingly, the development of *quasi-Newton methods* is motivated by the practical necessity of a method that achieves fast local convergence without using second-order derivatives. In a quasi-Newton method, an initial Hessian H_0 is usually defined as a multiple of the identity matrix, and then each successive approximate Hessian is obtained by a low-rank update to the previous approximate Hessian, i.e.,

$$H_{k+1} = H_k + U_k, \tag{2.31}$$

where U_k represents a low-rank matrix. The choice of U_k is based on the local affine model of ∇f near x_{k+1}:

$$\nabla f(x_k) \approx \nabla f(x_{k+1}) + \nabla^2 f(x_{k+1})(x_k - x_{k+1}).$$

Write $d_k = x_{k+1} - x_k$ and $w_k = \nabla f(x_{k+1}) - \nabla f(x_k)$, it follows that

$$w_k \approx \nabla^2 f(x_{k+1}) d_k.$$

The approximate Hessian H_{k+1} is defined such that the following condition holds:

$$w_k = H_{k+1} d_k$$

which is known as the *quasi-Newton condition* or *secant condition*. In addition, it is also crucial to retain the symmetry and positive-definiteness of each H_k. An updating formula (2.31) is said to have the property of *hereditary symmetry* if symmetry of H_k implies symmetry of H_{k+1}, and *hereditary positive-definiteness* if the positive-definiteness of H_k implies positive-definiteness of H_{k+1}. The well-known BFGS (Broyden-Fletcher-Goldfarb-Shanno) update

$$H_{k+1} = H_k - \frac{1}{d_k^{\mathrm{T}} H_k d_k} H_k d_k d_k^{\mathrm{T}} H_k + \frac{1}{w_k^{\mathrm{T}} d_k} w_k w_k^{\mathrm{T}} \tag{2.32}$$

is a symmetric, rank-two update to H_k that satisfies the quasi-Newton condition and has the property of hereditary symmetry. Moreover, if $w_k^{\mathrm{T}} d_k > 0$, then the BFGS update has the property of hereditary positive-definiteness. Indeed, the BFGS updating formula can be rewritten as

$$H_{k+1} = (I + v_k d_k^{\mathrm{T}}) H_k (I + d_k v_k^{\mathrm{T}}), \;\text{ with }\; v_k = \frac{1}{(w_k^{\mathrm{T}} d_k)^{1/2} (d_k^{\mathrm{T}} H_k d_k)^{1/2}} w_k - \frac{1}{d_k^{\mathrm{T}} H_k d_k} H_k d_k.$$

This identity implies that H_{k+1} is positive definite if H_k is positive definite and $I + d_k v_k^{\mathrm{T}}$ is nonsingular (i.e., if $1 + v_k^{\mathrm{T}} d_k \neq 0$). As H_{k+1} satisfies the quasi-Newton condition, it follows that

$$w_k^{\mathrm{T}} d_k = d_k^{\mathrm{T}} H_{k+1} d_k = d_k^{\mathrm{T}} (I + v_k d_k^{\mathrm{T}}) H_k (I + d_k v_k^{\mathrm{T}}) d_k = (1 + v_k^{\mathrm{T}} d_k)^2 d_k^{\mathrm{T}} H_k d_k.$$

It $w_k^{\mathrm{T}} d_k > 0$, then $(1 + v_k^{\mathrm{T}} d_k)^2 > 0$, which implies that $I + d_k v_k^{\mathrm{T}}$ is nonsingular. Therefore, to ensure that the approximate Hessian remains positive definite in a quasi-Newton method, the BFGS update is applied only when $w_k^{\mathrm{T}} d_k > 0$.

For a quasi-Newton method, an important advantage of using a line search based on the Wolfe conditions is that $w_k^{\mathrm{T}} d_k$ is always positive. This property is a consequence of the Wolfe step satisfying the inequality $\nabla f(x_{k+1})^{\mathrm{T}} p_k \geq \eta_W \nabla f(x_k^{\mathrm{T}} p_k$, which is implicitly imposed via the first Wolfe condition (2.11). The definition of $w_k^{\mathrm{T}} d_k$ yields

$$w_k^{\mathrm{T}} d_k = \alpha_k (\nabla f(x_{k+1})^{\mathrm{T}} p_k - \nabla f(x_k)^{\mathrm{T}} p_k \geq -\alpha_k (1 - \eta_W) \nabla f(x_k)^{\mathrm{T}} p_k > 0.$$

This property does not necessarily hold for an Armijo step.

The superlinear convergence of a quasi-Newton method can be proved assuming that the sequence of iterates converges. The proof of the following theorem can be found in [23].

Theorem 2.1.10 (Superlinear convergence of a quasi-Newton method [54])**.** *Let $f(x)$ be twice-continuously differentiable for all $x \in \mathbb{R}^n$. Assume that there exists $x^* \in \mathbb{R}^n$ such that $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite. Let $\{ H_k \}$ be a sequence of nonsingular matrices. Suppose that for some $x_0 \in \mathbb{R}^n$ the sequence of iterates $\{ x_k \}$ is given by*

$$x_{k+1} = x_k + p_k, \quad \textit{where} \;\; H_k p_k = -\nabla f(x_k).$$

If the sequence $\{ x_k \}$ converges to x^*, and that $x_k \neq x^*$ for any k, then $\{ x_k \}$ converges superlinearly to x^* if and only if

$$\lim_{k \to \infty} \frac{\|(H_k - \nabla^2 f(x^*)) p_k\|}{\|p_k\|} = 0.$$

□

For large-scale problems, limited-memory quasi-Newton methods are typically used. In contrast to the regular quasi-Newton methods, limited-memory methods require storage for only a few n-vectors.

For the conventional BFGS method, at the start of the k-th iteration, the inverse BFGS approximation may be written in the form

$$M_k = V_{k-1}^{\mathrm{T}} M_{k-1} V_{k-1} + \frac{1}{d_{k-1}^{\mathrm{T}} y_{k-1}} d_{k-1} d_{k-1}^{\mathrm{T}}, \;\; \text{with} \;\; V_{k-1} = I - \frac{1}{d_{k-1}^{\mathrm{T}} y_{k-1}} y_{k-1} d_{k-1}^{\mathrm{T}}.$$

A simple calculation gives $V_{k-1}^2 = V_{k-1} V_{k-1} = V_{k-1}$, which implies that V_{k-1} is an *oblique projection* that projects vectors onto the null-space of span(y_{k-1}), i.e., $V_{k-1} y_{k-1} = 0$.

If ρ_j denotes the quantity $\rho_j = 1/d_j^{\mathrm{T}} y_j$, then M_k can be written as

$$M_k = \big(V_{k-1}^{\mathrm{T}} V_{k-2}^{\mathrm{T}}\big) M_{k-2} \big(V_{k-2} V_{k-1}\big) + \rho_{k-2} V_{k-1}^{\mathrm{T}} d_{k-2} d_{k-2}^{\mathrm{T}} V_{k-1} + \rho_{k-1} d_{k-1} d_{k-1}^{\mathrm{T}}.$$

Continuing to expand each M_j in turn gives the update

$$\begin{aligned} M_k = {} & \big(V_{k-1}^{\mathrm{T}} \cdots V_{k-m}^{\mathrm{T}}\big) M_k^{(0)} \big(V_{k-m} \cdots V_{k-1}\big) \\ & + \rho_{k-m} \big(V_{k-1}^{\mathrm{T}} \cdots V_{k-m+1}^{\mathrm{T}}\big) d_{k-m} d_{k-m}^{\mathrm{T}} \big(V_{k-m+1} \cdots V_{k-1}\big) \\ & + \rho_{k-m+1} \big(V_{k-1}^{\mathrm{T}} \cdots V_{k-m+2}^{\mathrm{T}}\big) d_{k-m+1} d_{k-m+1}^{\mathrm{T}} \big(V_{k-m+2} \cdots V_{k-1}\big) \\ & \quad \vdots \\ & + \rho_{k-1} d_{k-1} d_{k-1}^{\mathrm{T}}. \end{aligned}$$

This formula implies that at the start of iteration k, a limited-memory variant of M_k may be defined by updating an initial matrix $M_k^{(0)}$ m times using the BFGS formula with the m pairs

$$(d_{k-1}, y_{k-1}), \quad (d_{k-2}, y_{k-2}), \quad \ldots, \quad (d_{k-m}, y_{k-m}).$$

The resulting matrix is used to define the search direction as $p_k = -M_k \nabla f(x_k)$. At the next step, (d_{k-m}, y_{k-m}) is discarded and the most recently computed pair (d_k, y_k) is added to the list of vector pairs. The total storage is $2m$ vectors.

Additionally, a limited-memory reduced-Hessian method will be reviewed later in Section 3.4.1.

2.2 Interior Methods

2.2.1 Inequality-constrained optimization

A nonlinear inequality-constrained optimization problem may be written in the general form

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) \geq 0, \tag{NIP}$$

where $c(x)$ is an m-vector of nonlinear constraint functions with ith components $c_i(x)$ $(i = 1, \cdots, m)$, and f and $\{\, c_i(x) \,\}$ are smooth functions that are assumed to be twice-continuously differentiable in this work. For a constrained optimization problem, any point x satisfying all the constraints is called a *feasible point*, and the set of all such points is the *feasible region*. The feasible region for this problem is given by

$$\Omega = \{\, x : c_i(x) \geq 0, \;\; i = 1, 2, \ldots, m \,\}.$$

Definition 2.2.1. *For problem (NIP), the constraint $c_i(x) \geq 0$ is said to be* satisfied *at $\bar{x}$ if $c(\bar{x}) \geq 0$,* active *if $c_i(\bar{x}) = 0$ and* inactive *if $c_i(\bar{x}) > 0$. The* active set *$\mathcal{A}(\bar{x})$ is the set of indices of the active constraints at $\bar{x}$, i.e., $\mathcal{A}(\bar{x}) = \{\, i : c_i(\bar{x}) = 0 \,\}$. The constraint $c_i(\bar{x}) \geq 0$ is said to be violated if $c_i(\bar{x}) < 0$.*

Definition 2.2.2 (First-order KKT point for (NIP))**.** *The* first-order KKT conditions *for the inequality-constrained problem* (NIP) *hold at the point x^*, or, equivalently, x^* is a (first-order)* KKT point*, if there exists an m-vector y^*, called a* Lagrange multiplier vector*, such that*

$$\begin{aligned} c(x^*) &\geq 0, && \text{(feasibility)} && \text{(2.33a)}\\ \nabla f(x^*) &= J(x^*)^{\mathrm{T}} y^*, && \text{(stationarity)} && \text{(2.33b)}\\ y^* &\geq 0, && \text{(nonnegativity of the multipliers)} && \text{(2.33c)}\\ c(x^*) \cdot y^* &= 0. && \text{(complementarity)} && \text{(2.33d)} \end{aligned}$$

The stationarity condition (2.33b) can be written as

$$\nabla_x L(x^*, y^*) = 0, \quad \text{where} \quad L(x, y) = f(x) - y^{\mathrm{T}} c(x). \tag{2.34}$$

Thus a KKT point is a stationary point with respect to x of the *Lagrangian function* $L(x, y)$ defined in (2.34). It is common to refer to x as the "primal variables" and to the Lagrange multipliers y as the "dual variables".

Definition 2.2.3 (Acceptable Lagrange multipliers)**.** *Given a KKT point x^* for problem* (NIP)*, the set of* acceptable Lagrange multipliers *is defined as*

$$\mathcal{Y}(x^*) = \{\, y \in \mathbb{R}^m : \nabla f(x^*) = J(x^*)^{\mathrm{T}} y, \;\; y \geq 0, \;\; \textit{and} \;\; c(x^*) \cdot y = 0 \,\}. \tag{2.35}$$

The complementarity condition $c(x^*) \cdot y = 0$ forces y_i to be zero if constraint i is inactive, but allows the possibility that $y_i = 0$ when constraint i is active. An important property, *strict complementarity*, occurs when all the multipliers for active constraints are positive.

Definition 2.2.4 (Strict complementarity)**.** Strict complementarity *holds at the KKT point x^* if there is a multiplier $y^* \in \mathcal{Y}$ such that $y_i^* > 0$ for all $i \in \mathcal{A}(x^*)$.*

Before considering the optimality conditions for (NIP), a formal definition of a constrained local minimizer is given as follows..

Definition 2.2.5 (Constrained local minimizer)**.** *Let f be a function defined for all $x \in \mathcal{D} \subseteq \mathbb{R}^n$. Let $\mathcal{N}(x^*, \delta)$ denote the set $\mathcal{B}(x^*, \delta) \cap \Omega$, where $\mathcal{B}(x^*, \delta)$ is an open ball centered at x^* with $\mathcal{B}(x^*, \delta) \subseteq \mathcal{D}$. A point x^* is a* constrained local minimizer *of f if there is a δ sufficiently small such that*

$$f(x^*) \leq f(x) \ \textit{ for all } x \in \mathcal{N}(x^*, \delta).$$

A point x^* is a strict *or* proper *constrained minimizer if this inequality holds with strict inequality except at x^* itself, i.e., $f(x^*) < f(x)$ for all $x \in \mathcal{N}(x^*, \delta)$.*

For problems with linear constraints, the first-order KKT conditions alone are *necessary* for optimality. However, to specify first-order necessary conditions for optimality with nonlinear constraints, the constraints are required to satisfy certain regularity conditions, known as *constraint qualifications*, at x^*. If these regularity conditions do not hold, a solution x^* may or may not be a KKT point. Two most commonly used constraint qualifications are defined below, which are collectively known as *first-order* constraint qualifications.

Definition 2.2.6 (linear independence constraint qualification)**.** *The* linear independence constraint qualification (LICQ) *holds at the feasible point $\bar{x}$ of (NIP) if x is strictly feasible, or if the active constraint gradients, $\{\, \nabla c_i(x) : i \in \mathcal{A}(x) \,\}$ at x are linearly independent, i.e., $J_{\mathrm{a}}(x)$ has full row rank.*

Definition 2.2.7 (Mangasarian-Fromovitz constraint qualification)**.** *The* Mangasarian-Fromovitz constraint qualification (MFCQ) *holds at the feasible point $\bar{x}$ of* (NIP) *if x is strictly feasible, or if there exists a vector p such that $\nabla c_i(x)^{\mathrm{T}} p > 0$ for all $i \in \mathcal{A}(x)$ (i.e., $J_{\mathrm{a}}(x)p > 0$).*

The MFCQ is a *weaker* condition than the LICQ in the sense that satisfaction of the LICQ implies the MFCQ, but not the reverse. An important consequence of the MFCQ is the boundedness of the set of acceptable multipliers.

Lemma 2.2.1 (Implication of the MFCQ: a bounded multiplier set [37])**.** *If $\bar{x}$ is a first-order KKT point at which the MFCQ is satisfied, then the set of multipliers $\mathcal{Y}$ defined in* (2.35) *is bounded.*

Proof. First we consider the nature of $\mathcal{Y}$ at $\bar{x}$, which consists of all $y \in \mathbb{R}^m$ satisfying $c(\bar{x}) \cdot y = 0$, $\nabla f(\bar{x}) = J(\bar{x})^{\mathrm{T}} y$, and $y \geq 0$. It is easy to see that $\mathcal{Y}(\bar{x})$ is convex. Given any $\bar{y} \in \mathcal{Y}(\bar{x})$, $\mathcal{Y}(\bar{x})$ can be unbounded only if there is a nonzero ray u emanating from $\bar{y}$ such that $\bar{y} + \alpha u \in \mathcal{Y}(\bar{x})$ for all

$\alpha \ge 0$. If such a ray exists, the complementarity condition will be satisfied only if components of u corresponding to inactive constraints are zero. Thus, in order for both $\bar{y}$ and $\bar{y} + \alpha u$ to lie in $\mathcal{Y}(\bar{x})$, it must be true that

$$\nabla f(\bar{x}) = J(\bar{x})^{\mathrm{T}} \bar{y}_{\mathrm{a}} = J_{\mathrm{a}}(\bar{x})^{\mathrm{T}} (\bar{y}_{\mathrm{a}} + \alpha u_{\mathrm{a}}),$$

where $\bar{y}_{\mathrm{a}}$ and u_{a} denote the subvectors of $\bar{y}$ and u corresponding to active constraints. It follows that $J_{\mathrm{a}}(\bar{x})^{\mathrm{T}} u_{\mathrm{a}} = 0$. Finally, $\bar{y}_{\mathrm{a}} + \alpha u_{\mathrm{a}}$ will remain nonnegative for arbitrarily large positive α only if $u_{\mathrm{a}} \ge 0$.

Turning now to the implications of the MFCQ, we know that a vector p exists such that $J_{\mathrm{a}}(\bar{x})p > 0$, which means that $\alpha J_{\mathrm{a}}(\bar{x})p > 0$ for any positive α. Thus for any positive θ there is a vector p satisfying $J_{\mathrm{a}}(\bar{x})p \ge \theta e$, where e denotes the vector of all ones. As a result, the linear program

$$\underset{p,\theta}{\text{maximize}}\ \theta \ \text{ subject to } \ J_{\mathrm{a}}(\bar{x})p - \theta e \ge 0, \ \ \theta \ge 0 \tag{2.36}$$

is feasible, but its objective function is unbounded above. Using standard duality theory for linear programming, unboundedness of the primal objective implies infeasibility of the dual. The constraints of the dual corresponding to (2.36) are $J_{\mathrm{a}}(\bar{x})^{\mathrm{T}} u_{\mathrm{a}} = 0$, $e^{\mathrm{T}} u_{\mathrm{a}} = 1$, and $u_{\mathrm{a}} \ge 0$, and so we know that there is no vector u_{a} satisfying these conditions. But, as shown in the first part of the proof, these are precisely the properties that u_{a} must have in order for $\mathcal{Y}(\bar{x})$ to be unbounded. (The condition $e^{\mathrm{T}} u_{\mathrm{a}} = 1$ is simply a scaling restriction to ensure that $u_{\mathrm{a}} \ne 0$.) Consequently no ray u exists, and $\mathcal{Y}(\bar{x})$ is bounded. □

A practical disadvantage of the MFCQ compared to the LICQ is that verifying whether the MFCQ is satisfied is more difficult—in fact, determining whether or not the MFCQ holds requires solving a linear program. The argument developed in the proof of Lemma 2.2.1 shows that the MFCQ holds at the KKT point $\bar{x}$ if the optimal solution of the linear program

$$\underset{p,\theta}{\text{maximize}}\ \theta \quad \text{subject to} \quad J_{\mathrm{a}}(\bar{x})p - \theta e \ge 0, \ \ 0 \le \theta \le 1, \tag{2.37}$$

occurs at the maximum possible value of θ, namely $\theta = 1$. Note that this linear programming (LP) problem is feasible because its constraints are satisfied by $\theta = 0$ and $p = 0$.

The main first-order necessary condition for a solution of problem (NIP) can now be stated.

Theorem 2.2.1 (First-order necessary conditions). *Let x^* be a point such that $c(x^*) \ge 0$, with $c_{\mathrm{a}}(x) = 0$. If MFCQ holds at x^*, then x^* is a local minimizer of* (NIP) *only if x^* is a first-order*

KKT point, i.e., there exists a vector $y_{\rm a}^*$ *such that*

$$\nabla f(x^*) = J_{\rm a}(x^*)^{\rm T} y_{\rm a}^*, \;\; \textit{with} \;\; y_{\rm a}^* \geq 0. \tag{2.38}$$

Proof. See [37, Lemma 2.16]. □

For *second-order* optimality conditions, the Hessian of the Lagrangian $L(x, y)$ (2.34) with respect to x,

$$H(x, y) = \nabla^2{}_{xx} L(x, y) = \nabla^2 f(x) - \sum_{i=1}^{m} y_i \nabla^2 c_i(x),$$

plays a crucial role. To make explicit use of the information about the stationarity of the objective function, second-order conditions typically involve curvature of the Lagrangian function along feasible directions in the set

$$\mathcal{C}_L(x) = \big\{\, p : p \neq 0, \, \nabla f(x)^{\rm T} p = 0 \text{ and } J_{\rm a}(x) p \geq 0 \,\big\},$$

which is known as the *critical cone.*

In order to formulate an appropriate second-order constraint qualification, it is useful to write the set $\mathcal{C}_L(x)$ in an equivalent form that requires x to be a KKT point, i.e., the set of acceptable multipliers is not empty at x. At any KKT point x, choose some $y \in \mathcal{Y}(x)$ and let $\mathcal{A}_+(x, y)$ denote the set of indices of active constraints with *positive* Lagrange multipliers and let $J_+(x)$ denote the corresponding matrix of constraint gradients. Similarly, let $\mathcal{A}_0(x, y)$ denote the set of indices of active constraints with zero multipliers, and let $J_0(x)$ denote the associated matrix of constraint gradients. Note that $\mathcal{A}_0(x, y)$ and $\mathcal{A}_+(x, y)$ define a complete partition of the active set, i.e., $\mathcal{A}(x) = \mathcal{A}_0(x, y) \cup \mathcal{A}_+(x, y)$. Moreover, this partition is a function of the particular $y \in \mathcal{Y}(x)$. If y_+ and y_0 denote the vectors of positive and zero components of y, it follows that

$$\nabla f(x)^{\rm T} p = y^{\rm T} J(x) p = y_+^{\rm T} J_+(x) p + y_0^{\rm T} J_0(x) p = y_+^{\rm T} J_+(x) p = 0.$$

This implies that the critical cone may be written in the form

$$\mathcal{C}_L(x) = \big\{\, p : p \neq 0 \text{ such that } J_+(x) p = 0 \text{ and } J_0(x) p \geq 0 \,\big\}.$$

This characterization suggests the use of an appropriate "second-order" tangent cone defined in terms of the functions $c_0(x)$ and $c_+(x)$ given by the elements of $c(x)$ with indices in $\mathcal{A}_0(x, y)$ and $\mathcal{A}_+(x, y)$ respectively.

Definition 2.2.8 (Second-order constraint qualification (SOCQ))**.** *The second-order constraint qualification for inequality constraints holds at a KKT point x if, for all $y \in \mathcal{Y}(x)$, every nonzero p satisfying $J_+(x)p = 0$ and $J_0(x)p \ge 0$ is tangent to a twice-differentiable path $x(\alpha)$ such that $c_+\big(x(\alpha)\big) = 0$ and $c_0\big(x(\alpha)\big) \ge 0$ for all $0 < \alpha \le \widehat{\alpha}$.*

Theorem 2.2.2 (Second-order necessary conditions for (NIP) [54])**.** *If the first- and second-order constraint qualifications hold at x^*, then x^* is a local solution of* (NIP) *only if*

(**a**) *x^* is a KKT point, i.e., $c(x^*) \ge 0$ and there exists a nonempty set $\mathcal{Y}(x^*)$ of multipliers y satisfying $y \ge 0$, $c(x^*) \cdot y = 0$, and $\nabla f(x^*) = J(x^*)^{\mathrm{T}} y$;*

(**b**) *for some $y \in \mathcal{Y}(x^*)$ and all $p \ne 0$ satisfying $\nabla f(x^*)^{\mathrm{T}} p = 0$ and $J_{\mathrm{a}}(x^*)p \ge 0$, it holds that $p^{\mathrm{T}} H(x^*, y)p \ge 0$.*

Proof. Part **(a)** follows immediately from Theorem 2.2.1. To prove part **(b)**, consider any nonzero vector p satisfying $\nabla f(x^*)^{\mathrm{T}} p = 0$ and $J_{\mathrm{a}}(x^*)p \ge 0$. Because of the second-order constraint qualification, p is tangent to a twice-differentiable feasible path $x^*(\alpha)$ such that $c_+\big(x^*(\alpha)\big) = 0$ and $c_0\big(x^*(\alpha)\big) \ge 0$ for all $0 < \alpha \le \widehat{\alpha}$, with $x^*(0) = x^*$. Let v denote $d^2 x^*(\alpha)/d\alpha^2\big|_{\alpha=0}$, and assume henceforth that all vector and matrix functions are evaluated at x^* unless otherwise specified. As for each $i \in \mathcal{A}_+(x, y)$, the constraint function c_i is identically zero along $x^*(\alpha)$, we have

$$\begin{aligned} \frac{d^2}{d\alpha^2} c_i\big(x^*(\alpha)\big)\Big|_{\alpha=0} &= \nabla c_i\big(x^*(\alpha)\big)^{\mathrm{T}} \frac{d^2}{d\alpha^2} x^*(\alpha)\Big|_{\alpha=0} + \frac{d}{d\alpha}\big(\nabla c_i\big(x^*(\alpha)\big)^{\mathrm{T}}\big) \frac{d}{d\alpha} x^*(\alpha)\Big|_{\alpha=0} \\ &= \nabla c_i(x^*)^{\mathrm{T}} v + p^{\mathrm{T}} \nabla^2 c_i(x^*) p = 0. \end{aligned} \tag{2.39}$$

Further, using the expression $\nabla f(x^*) = J(x^*)^{\mathrm{T}} y^* = J_+(x^*)^{\mathrm{T}} y_+^*$ from (2.33) and the assumption that $J_+(x^*)p = 0$,

$$\frac{d}{d\alpha} f\big(x^*(\alpha)\big)\Big|_{\alpha=0} = \nabla f\big(x^*(\alpha)\big)^{\mathrm{T}} \frac{d}{d\alpha} x^*(\alpha)\Big|_{\alpha=0} = \nabla f(x^*)^{\mathrm{T}} p = y_+^{*\mathrm{T}} J_+(x^*)p = 0. \tag{2.40}$$

As a KKT point, x^* is a stationary point of f along the feasible path. In order for x^* to be a local solution, the *curvature* of f along any feasible path must be nonnegative, i.e., it must hold that

$$\frac{d^2}{d\alpha^2} f\big(x^*(\alpha)\big)\Big|_{\alpha=0} \ge 0. \tag{2.41}$$

Using (2.40) and the definition of v, we write (2.41) as

$$\begin{aligned}\frac{d^2}{d\alpha^2} f\big(x^*(\alpha)\big)\bigg|_{\alpha=0} &= \frac{d}{d\alpha}\Big(\nabla f\big(x^*(\alpha)\big)^{\mathrm{T}} \frac{d}{d\alpha} x^*(\alpha)\Big)\bigg|_{\alpha=0} \\ &= \nabla f(x^*)^{\mathrm{T}} \;\frac{d^2}{d\alpha^2} x^*(\alpha)\bigg|_{\alpha=0} + p^{\mathrm{T}} \nabla^2 f(x^*) p \\ &= y_+^{*\mathrm{T}} J_+(x^*) v + p^{\mathrm{T}} \nabla^2 f(x^*) p. \end{aligned} \tag{2.42}$$

Rewriting (2.39) as $\nabla c_i(x^*)^{\mathrm{T}} v = -p^{\mathrm{T}} \nabla^2 c_i(x^*) p$ for $i \in \mathcal{A}_+(x, y)$, and substituting this expression into (2.42), we obtain

$$\frac{d^2}{d\alpha^2} f\big(x^*(\alpha)\big)\bigg|_{\alpha=0} = -p^{\mathrm{T}}\Big(\sum_{i=1}^{m} y_i^* \nabla^2 c_i(x^*)\Big) p + p^{\mathrm{T}} \nabla^2 f(x^*) p = p^{\mathrm{T}} H(x^*, y^*) p \geq 0,$$

where $H(x^*, y^*)$ is the Hessian of the Lagrangian at $(x, y) = (x^*, y^*)$. □

If the active constraints are linear, then both the first- and second-order constraint qualifications hold. Similarly, the LICQ is sufficient to ensure that both the first- and second-order constraint qualifications to hold.

Definition 2.2.9 (Isolated constrained minimizer)**.** *A local constrained minimizer x^* is isolated if there's a neighborhood of x^* containing no other local constrained minimizers.*

Theorem 2.2.3 (Sufficient conditions for an isolated minimizer)**.** *A point x^* is an isolated local constrained minimizer of* (NIP) *if*

(**a**) *x^* is a KKT point, i.e., $c(x^*) \geq 0$ and there exists a nonempty set $\mathcal{Y}$ of multipliers y satisfying $y \geq 0$, $c(x^*) \cdot y = 0$, and $\nabla f(x^*) = J(x^*)^{\mathrm{T}} y$;*

(**b**) *the MFCQ holds at x^*, i.e., there is a vector p such that $J_{\mathrm{a}}(x^*) p > 0$;*

(**c**) *for all $y \in \mathcal{Y}$ and all nonzero p satisfying $\nabla f(x^*)^{\mathrm{T}} p = 0$ and $J_{\mathrm{a}}(x^*) p \geq 0$, there exists $\omega > 0$ such that $p^{\mathrm{T}} H(x^*, y) p \geq \omega \|p\|^2$.*

Proof. See [37, Theorem 2.23]. □

Although Theorem 2.2.3 is very nice, its conditions are not easy to check in their full generality. The verification of assumption (**c**) for all p such that $\nabla f(x^*)^{\mathrm{T}} p = 0$ and $J_{\mathrm{a}}(x^*) p \geq 0$ requires finding the global minimizer of a possibly indefinite quadratic form over a cone, an NP-hard problem, not to mention the issue of how to check that (**c**) holds for *all* $y \in \mathcal{Y}(x^*)$. If, however, the gradients of the active constraints at x^* are linearly independent and strict complementarity holds, Theorem 2.2.3 leads immediately to the following result.

Theorem 2.2.4 (Strong sufficient conditions for an isolated minimizer)**.** *A point x^* is an isolated local constrained minimizer of* (NIP) *if*

(**a**) *the LICQ holds at x^*, i.e., $J_{\mathrm{a}}(x^*)$ has full row rank;*

(**b**) *x^* is a KKT point and strict complementarity holds, i.e., the (necessarily unique) multiplier y^* has the property that $y_i^* > 0$ for all $i \in \mathcal{A}(x^*)$;*

(**c**) *for all nonzero p satisfying $J_{\mathrm{a}}(x^*)p = 0$, there exists an $\omega > 0$ such that*

$$p^{\mathrm{T}} H(x^*, y^*)p \geq \omega \|p\|^2.$$

Proof. See [37, Theorem 2.24]. □

As an alternative, many methods for inequality constraints may be motivated by considering a form of (NIP) in which the nonlinear inequality constraints are converted to equalities using a set of nonnegative *slack variables*. A slack variable s_i can be used to convert the inequality constraint $c_i(x) \geq 0$ to an equality constraint by means of the transformation:

$$c_i(x) \geq 0 \quad \text{if and only if} \quad c_i(x) - s_i = 0, \quad s_i \geq 0.$$

This gives the following mixed-constraint problem, which is equivalent to (NIP):

$$\underset{x\in\mathbb{R}^n, s\in\mathbb{R}^m}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) - s = 0, \quad s \geq 0, \qquad \text{(NIPs)}$$

Note that problem (NIPs) has two types of constraint: the nonlinear equality constraints $c(x) - s = 0$ and the nonnegativity constraints $s \geq 0$. The treatment of equality constraints will be discussed in Section 2.2.4. Problems of the form (NIPs) are especially convenient when the x-variables are already subject to upper and lower bounds, which is a very common situation in practice. In this situation, it is best to treat the bounds specially and not include them in the general inequalities $c(x) \geq 0$.

2.2.2 Barrier methods

Classical barrier methods for inequality-constrained optimization is the foundation of the modern interior methods. A *barrier* method is motivated by the unconstrained minimization of a function that combines the original objective function with a positively weighted barrier term that prevents iterates from leaving the feasible region. The predominant barrier function today is the *logarithmic barrier function*:

$$B(x\,;\mu) = f(x) - \mu \sum_{i=1}^{m} \ln c_i(x), \tag{2.43}$$

where μ is a positive scalar that is known as the *barrier parameter*. An important feature of $B(x,\mu)$ is that it retains the smoothness properties of $f(x)$ and $c(x)$ as long as $c(x) > 0$.

Minimizers of the barrier function and local constrained minimizers of the original problem are closely related. An initial hint of those relationships can be seen algebraically from the gradient of the barrier function (2.43), denoted by $\nabla B(x\,;\mu)$, which can be expressed in various equivalent forms:

$$\begin{aligned} \nabla B(x\,;\mu) &= \nabla f(x) - \sum_{i=1}^{m} \frac{\mu}{c_i(x)}\, \nabla c_i(x), && \text{(2.44a)}\\ &= \nabla f(x) - \mu J(x)^{\mathrm{T}} C(x)^{-1} e, \ \text{ and} && \text{(2.44b)}\\ &= \nabla f(x) - J(x)^{\mathrm{T}} \big(\mu \mathbin{\cdot/} c(x)\big). && \text{(2.44c)} \end{aligned}$$

In the form (2.44b), $C(x)$ denotes the $m \times m$ diagonal matrix of constraint values and e the m-vector of all ones.

An unconstrained minimizer of $B(x\,;\mu)$ will be denoted by either x_μ or $x(\mu)$, and it will be shown that $c(x_\mu) > 0$. Because $\nabla B(x\,;\mu)$ is twice-continuously differentiable, it must hold that

$$\nabla B(x_\mu\,;\mu) = \nabla f(x_\mu) - \mu J(x_\mu)^{\mathrm{T}} C(x_\mu)^{-1} e = 0. \tag{2.45}$$

It follows that

$$\nabla f(x_\mu) = \mu J(x_\mu)^{\mathrm{T}} C(x_\mu)^{-1} e = \sum_{i=1}^{m} \frac{\mu}{c_i(x_\mu)}\, \nabla c_i(x_\mu).$$

Hence the objective gradient at x_μ is a positive linear combination of the constraint gradients. The coefficients in that linear combination are called the *barrier multipliers* (by analogy with Lagrange multipliers) and denoted by y_μ. Formally, y_μ is defined as

$$y_\mu = \mu C(x_\mu)^{-1} e = \mu \mathbin{\cdot/} c(x_\mu).$$

Thus the gradient at x_μ can be expressed as

$$\nabla f(x_\mu) = \sum_{i=1}^{m} \frac{\mu}{c_i(x_\mu)} \, \nabla c_i(x_\mu) = J(x_\mu)^{\mathrm{T}} y_\mu, \text{ with } y_\mu > 0,$$

which resembles the stationarity and nonnegativity properties (2.33b) and (2.33c), $\nabla f(x^*) = J(x^*)^{\mathrm{T}} y^*$ and $y^* \geq 0$, that hold at a KKT point. Moreover, the definition of y_μ implies that

$$c(x_\mu) \cdot y_\mu = \mu, \text{ or } c_i(x_\mu)(y_\mu)_i = \mu, \;\; i = 1, \ldots, m. \tag{2.46}$$

This componentwise relationship between the barrier multipliers, constraint values, and the barrier parameter, called *perturbed complementarity*, is analogous as $\mu \to 0$ to the complementarity condition $c(x^*) \cdot y^* = 0$ (2.33d) that holds at a KKT point.

An alternative interpretation of $\nabla B(x\,;\mu) = 0$ can be derived by defining m new independent variables y and writing (2.45) as $n + m$ nonlinear equations in x and y:

$$\begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y \\ C(x)y - \mu e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{2.47}$$

Recalling that the barrier multipliers y_μ are defined as $\mu \cdot\!/\; c(x_\mu)$, it is easy to see that (x_μ, y_μ) satisfy the nonlinear equations (2.47). Conversely, given any solution $(x\,; y)$ of (2.47), the associated vector x is a stationary point of the barrier function with parameter μ. The equations (2.47) are called the *perturbed optimality conditions.*

Definition 2.2.10 (Strictly feasible points)**.** *The subset of points in Ω for which all the constraint functions are* strictly positive *is denoted by* $\operatorname{int}_c(\Omega)$ *and defined as*

$$\operatorname{int}_c(\Omega) = \big\{\, x : c_i(x) > 0, \quad i = 1, \ldots, m \,\big\}.$$

A point x in $\operatorname{int}_c(\Omega)$ *is said to be* strictly feasible*.*

A general convergence theorem for the barrier methods (Theorem 2.2.5) depends on the existence of a subset of local constrained minimizers that is "isolated" within the full set of local constrained minimizers.

Definition 2.2.11 (Isolated subset)**.** *Given sets $\mathcal{X}^* \subseteq \mathcal{X} \subseteq \mathbb{R}^n$, $\mathcal{X}^*$ is an* isolated *subset of $\mathcal{X}$ if there exists a closed set $\mathcal{E}$ such that $\mathcal{X}^* \subset \operatorname{int}(\mathcal{E})$ and $\mathcal{E} \cap \mathcal{X} = \mathcal{X}^*$.*

The following convergence theorem requires that at least one of the points in $\mathcal{X}^*$ must lie in the closure of $\mathrm{int}_c(\mathit{\Omega})$, i.e., $\mathcal{X}^*$ contains either a strictly feasible point or a limit point of $\mathrm{int}_c(\mathit{\Omega})$. This assumption is needed because barrier methods can be viewed as finding the infimum of f subject to $c(x) > 0$, so that a sequence of barrier minimizers, each of which lies in $\mathrm{int}_c(\mathit{\Omega})$, can only converge to a feasible point that lies within the closure of $\mathrm{int}_c(\mathit{\Omega})$.

Theorem 2.2.5 (Local convergence for barrier methods)**.** *Consider the problem of minimizing $f(x)$ subject to $c(x) \geq 0$, where f and c are continuous. Let $\mathit{\Omega}$ denote the feasible region, let $\mathcal{X}$ denote the set of minimizers with objective function value f^*, and assume that $\mathcal{X}$ is nonempty. Let $\{\, \mu_k \,\}$ be a strictly decreasing sequence of positive barrier parameters such that $\lim_{k\to\infty} \mu_k = 0$. Assume that*

(A1) *there exists a nonempty compact set $\mathcal{X}^*$ of local minimizers that is an isolated subset of $\mathcal{X}$;*

(A2) *at least one point in $\mathcal{X}^*$ lies in the closure of $\mathrm{int}_c(\mathit{\Omega})$.*

Then the following results hold:

(i) *there exists a compact set $\mathcal{S}$ such that $\mathcal{X}^* \subset \mathrm{int}(\mathcal{S})$ and such that, for any feasible point $\bar{x}$ in $\mathcal{S}$ but not in $\mathcal{X}^*$, $f(\bar{x}) > f^*$;*

(ii) *for all sufficiently small μ_k, there is an* unconstrained *minimizer w_k of the barrier function $B(x\,;\mu_k)$ in $\mathrm{int}_c(\mathit{\Omega}) \cap \mathrm{int}(\mathcal{S})$, with*

$$B(w_k\,;\mu_k) = \min\{\, B(x\,;\mu_k) : x \in \mathrm{int}_c(\mathit{\Omega}) \cap \mathcal{S} \,\}.$$

Thus $B(w_k\,;\mu_k)$ is the smallest value of $B(x\,;\mu_k)$ for any $x \in \mathrm{int}_c(\mathit{\Omega}) \cap \mathcal{S}$;

(iii) *any sequence of these unconstrained minimizers $\{\, w_k \,\}$ of $B(x\,;\mu_k)$ has at least one convergent subsequence;*

(iv) *the limit point x_∞ of any convergent subsequence $\{\, x_k \,\}$ of the unconstrained minimizers $\{\, w_k \,\}$ defined in* (ii) *lies in $\mathcal{X}^*$;*

(v) *for the convergent subsequences $\{\, x_k \,\}$ of part* (iv)*,*

$$\lim_{k\to\infty} f(x_k) = f^* = \lim_{k\to\infty} B(x_k\,;\mu_k).$$

Proof. See [37, Theorem 3.10]. □

The following theoerem summarizes conditions under which a sequence of barrier minimizers not only converges to x^* but also defines a differentiable path to x^*. Depending on the context, this path is called either the *central path* or the *barrier trajectory.*

Theorem 2.2.6 (Properties of the central path/barrier trajectory)**.** *Consider the problem of minimizing $f(x)$ subject to $c(x) \geq 0$. Let Ω denote the feasible region, and assume that the set* int_c *of strictly feasible points is nonempty. Let x^* be a local constrained minimizer. Assume that the following sufficient optimality conditions hold at x^*:*

(**a**) *x^* is a KKT point, i.e., there exists a nonempty set $\mathcal{Y}$ of Lagrange multipliers y satisfying*

$$\mathcal{Y} = \left\{\, y : \nabla f(x^*) = J(x^*)^{\mathrm{T}} y, \;\; y \geq 0, \;\; \textit{and} \;\; c(x^*) \cdot y = 0 \,\right\};$$

(**b**) *the MFCQ* (Definition 2.2.7) *holds at x^*, i.e., there exists p such that $J_{\mathrm{a}}(x^*)p > 0$, where $J_{\mathrm{a}}(x^*)$ denotes the Jacobian of the active constraints at x^*; and*

(**c**) *there exists $\omega > 0$ such that $p^{\mathrm{T}} H(x^*, y) p \geq \omega \|p\|^2$ for all $y \in \mathcal{Y}$ and all nonzero p satisfying $\nabla f(x^*)^{\mathrm{T}} p = 0$ and $J_{\mathrm{a}}(x^*)p \geq 0$, where $H(x^*, y)$ is the Hessian of the Lagrangian.*

Assume that a logarithmic barrier method is applied in which μ_k converges monotonically to zero as $k \to \infty$. Then

(i) *there is at least one subsequence of unconstrained minimizers of the barrier function $B(x\,; \mu_k)$ converging to x^*;*

(ii) *if $\{\, x^k \,\}$ is one such a convergent subsequence, then the sequence of barrier multipliers $\{\, y^k \,\}$, whose ith component is $\mu_k / c_i(x^k)$, is bounded;*

(iii) $\lim_{k\to\infty} y^k = \bar{y} \in \mathcal{Y}$.

If, in addition, strict complementarity holds at x^*, i.e., there is a vector $y \in \mathcal{Y}$ such that $y_i > 0$ for all $i \in \mathcal{A}$, then

(iv) $\bar{y}_{\mathrm{a}} > 0$*;*

(v) *for sufficiently large k, the Hessian matrix $\nabla^2 B(x^k, \mu_k)$ is positive definite;*

(vi) *a unique, continuously differentiable vector function $x(\mu)$ of unconstrained minimizers of $B(x\,; \mu)$ exists for positive μ in a neighborhood of $\mu = 0$; and*

(vii) $\lim_{\mu \to 0_+} x(\mu) = x^*$.

Proof. See [37, Theorem 3.12]. □

If the strict complementarity holds at x^*, a useful corollary can be derived.

Corollary 2.2.1. *Under assumptions* (a)–(c) *of Theorem* 2.2.6 *and the added assumption of strict complementarity at* x^*, $\|x^k - x^*\| = \Theta(\mu_k)$.

Proof. See [37, Corollary 3.14]. □

A direct translation of the above theory into practice leads to a method in which minimizers x_μ of the barrier function are computed for a sequence of positive barrier parameters μ converging to zero. Such a method is structured into inner and outer iterations, where the inner iterations apply a line-search method or a trust-region method to compute an unconstrained minimizer of $B(x\,;\mu)$ for a fixed value of μ, and the outer iterations test for convergence and adjust μ. In recent algorithms, the idea is to improve efficiency by performing only an inexact minimization of the barrier function for each particular μ. With such a strategy, inner iterations are executed until a suitable measure of improvement has been achieved; the barrier parameter is then reduced and the process repeated.

While a general-purpose unconstrained technique such as a Newton-based line-search method can be applied to solve the subproblem of minimizing the barrier function $B(x\,;\mu)$, it should be noted that there is always an *implicit* constraint $c(x) > 0$ as $B(x\,;\mu)$ is only well-defined at strictly feasible points. For linear constraints, often a "fraction to the boundary" parameter is used such that the initial step is taken as a fraction of the distance to the boundary, thereby retaining the strict feasibility of the next iterate. For nonlinear constraints, the determination of the step to the boundary may require additional evaluations of the constraint functions. Moreover, as many general-purpose line-search techniques rely on polynomial interpolations, they may not be well suited to the extreme behavior of barrier functions near the boundary. Various special-purpose line searches have been proposed for use in barrier methods (see e.g., MurW94).

Given an interior point x, the *classical Newton barrier equations* $\nabla^2 B(x\,;\mu) = -\nabla B(x\,;\mu)$ at the current point x are

$$\Big(\nabla^2 f(x) - \sum_{i=1}^{m} \frac{\mu}{c_i(x)} \nabla^2 c_i(x) + \mu J(x)^{\mathrm{T}} C(x)^{-2} J(x)\Big) p = -\nabla f(x) + \mu J(x)^{\mathrm{T}} C(x)^{-1} e. \tag{2.48}$$

The equations (2.48) may be simplified by introducing an auxiliary m-vector $\pi(x\,;\mu)$, which can be viewed as a Lagrange multiplier estimate defined at an arbitrary strictly feasible point.

Definition 2.2.12 (Primal multipliers). *At any strictly feasible point x, the vector $\pi(x\,;\mu)$ with components $\mu/c_i(x)$ is known as the vector of* primal multipliers*. The dependence of π on μ may be omitted if μ is obvious.*

For any sequence $\{\,x_k\,\}$ converging to x_μ, it must hold that $\lim_{k\to\infty}\pi(x_k,\mu) = y_\mu$. Substituting the vector π in (2.48), the barrier gradient and Hessian may be written in the form

$$\begin{aligned}\nabla B(x\,;\mu) &= \nabla f(x) - J(x)^{\mathrm{T}}\pi(x\,;\mu) \;\text{ and}\\ \nabla^2 B(x\,;\mu) &= H\bigl(x,\pi\bigr) + J(x)^{\mathrm{T}}\,\mathit{\Pi}(x,\mu)\,C^{-1}(x)\,J(x),\end{aligned} \tag{2.49}$$

where $H(x,\pi)$ is the Hessian of the Lagrangian evaluated with $y=\pi$, and $\mathit{\Pi}$ is the matrix diag(π_1, π_2, ..., π_m). These relations indicate that the barrier derivatives are intimately related to those of the Lagrangian evaluated with the primal multipliers.

It is well-known that the Hessian matrix $\nabla^2 B$ becomes increasingly ill-conditioned as $\mu\to 0$ (see e.g., [88]). Thanks to a fortuitous combination of the special structure of the linear system and the cancellation errors that arise in computation, it is usually possible to solve the Newton barrier equations with acceptable accuracy despite the ill-conditioning (see [89]). However, it can be shown that the *exact* Newton step is inherently flawed in minimizing the classical barrier function because of the strong possibility of violating the constraints and its inefficiency in following the barrier trajectory (see [37]).

As an extension of the classical barrier methods, *modified barrier methods* [8, 15, 56, 72, 77] define a sequence of unconstrained problems in which the value of μ remains *bounded away from zero*, thereby avoiding the need to solve a problem whose Hessian becomes extremely ill-conditioned as μ approaches zero. The modified barrier methods are based on the observation that for a fixed positive μ, the constraints $c_i(x)\ge 0$ and $\mu\ln\bigl(1+c_i(x)/\mu\bigr)\ge 0$ are equivalent, i.e., their associated sets of feasible points are identical. Moreover, a KKT point for the original problem (NIP) is also a KKT point for the modified problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\;\; f(x) \quad \text{subject to} \;\; \mu\ln\bigl(1+c_i(x)/\mu\bigr)\ge 0,\;\; i=1,2,\ldots,m. \tag{2.50}$$

This motivates the definition of the *modified barrier function*:

$$M(x,y\,;\mu) = f(x) - \mu\sum_{i=1}^{m} y_i\ln\bigl(1+c_i(x)/\mu\bigr), \tag{2.51}$$

which can be interpreted as the conventional Lagrangian function for the modified problem (2.50).

The modified barrier function implicitly imposes the *shifted* constraints $c_i(x) + \mu \geq 0$, $i = 1, 2, \ldots, m$. Therefore, the barrier parameter μ can be alternatively interpreted as shifts for the constraints ([15]).

A complete theory analogous to that of the classical logarithmic barrier function exists for the modified barrier function (see [77]). A crucial property of the modified barrier function is that if y^* is a multiplier vector in $\mathcal{Y}(x^*)$ (see Definition 2.2.3), then there exists a fixed μ^* such that for all $\mu < \mu^*$, the corresponding x^* is a local minimizer of $M(x, y^* \,; \mu)$, i.e., $\nabla M(x^*, y^* \,; \mu) = 0$ and $\nabla^2 M(x^*, y^* \,; \mu)$ is positive semidefinite. It follows that, if an optimal multiplier is known, x^* can be found from just *one* unconstrained minimization.

In practice, neither the optimal multiplier vector nor an upper bound on μ is known in advance. As a result, a sequence of problems must be solved in which each $M(x, y \,; \mu)$ is defined with estimates of y^* and μ^*. The multiplier estimate is updated following each subproblem, and the barrier parameter is reduced if $\nabla^2 M(x, y \,; \mu)$ is not sufficiently positive definite. (For details, see, e.g., [8, 56, 72, 77].)

2.2.3 Primal-dual interior methods

Due to inherent flaws in the classical barrier method, it is desirable to develop interior methods that retain the good properties of the classical methods while avoiding their defects. As a result, primal-dual interior methods based on properties of $x(\mu)$ become increasingly popular for solving general nonlinear programming problems (see e.g., [11, 16, 26, 34, 42, 80, 82]). In a primal-dual method, the original (primal) variables x and the dual variables y (representing the Lagrange multipliers) are treated as independent.

The usual motivation for primal-dual methods is to find (x, y) satisfying the equations that hold at $x(\mu)$. Based on the perturbed optimality conditions (2.47), the goal is to compute a feasible solution $(x(\mu), y(\mu))$ of the $n + m$ nonlinear equations $F^\mu(x, y) = 0$, where

$$F^\mu(x, y) = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y \\ C(x) y - \mu e \end{pmatrix}.$$

Let v denote the $(n + m)$-vector of the combined unknowns (x, y) at a point that is strictly feasible in both x and y, i.e., $c(x) > 0$ and $y > 0$. If $F^\mu(v)$ denotes the function $F^\mu(x, y)$, then a Newton direction $\mathit{\Delta} v = (\mathit{\Delta} x, \mathit{\Delta} y)$ is defined by the Newton equations $F^\mu(v)' \mathit{\Delta} v = -F^\mu(v)$. After collecting terms on the right-hand side, the Newton *primal-dual equations* may be expressed as

$$\begin{pmatrix} H(x,y) & -J(x)^{\mathrm{T}} \\ YJ(x) & C(x) \end{pmatrix} \begin{pmatrix} \mathit{\Delta} x \\ \mathit{\Delta} y \end{pmatrix} = - \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y \\ C(x)\big(y - \pi(x,\mu)\big) \end{pmatrix}, \tag{2.52}$$

where $H(x,y)$ is the Hessian of the Lagrangian evaluated at (x,y).

The success of primal-dual methods is due in part to their effectiveness at following the barrier trajectory. In particular, if $(x,y) = \big(x(\mu), y(\mu)\big)$ is a point on the trajectory, and the barrier parameter is reduced from μ to $\widehat{\mu}$, the primal-dual direction (i.e., the solution of (2.52) with $\mu = \widehat{\mu}$) is *tangent* to the trajectory at (x,y). This property is easily shown by noting that on the trajectory, the relations $y = \pi(x,\mu)$ and $\nabla f(x) - J(x)^{\mathrm{T}} y = \nabla f(x) - J(x)^{\mathrm{T}} \pi = 0$ hold, and hence $\mathit{\Delta} x$ and $\mathit{\Delta} y$ satisfy

$$\begin{pmatrix} H(x,y) & -J(x)^{\mathrm{T}} \\ YJ(x) & C(x) \end{pmatrix} \begin{pmatrix} \mathit{\Delta} x \\ \mathit{\Delta} y \end{pmatrix} = - \begin{pmatrix} 0 \\ \big(\mu - \widehat{\mu}\big)e \end{pmatrix}, \tag{2.53}$$

On the other hand, differentiating $F^{\mu}(x,y) = 0$ with respect to μ leads to the following equations for $\big(x'(\mu), y'(\mu)\big)$:

$$\begin{pmatrix} H\big(x(\mu), y(\mu)\big) & -J(x(\mu))^{\mathrm{T}} \\ Y(\mu)J(x(\mu)) & C(x(\mu)) \end{pmatrix} \begin{pmatrix} x'(\mu) \\ y'(\mu) \end{pmatrix} = \begin{pmatrix} 0 \\ e \end{pmatrix}. \tag{2.54}$$

If the strong sufficient optimality conditions of Theorem 2.2.4 hold at x^*, then the matrix of (2.54) has a bounded condition number as $\mu \to 0$. Comparing (2.53) with (2.54) shows that $\mathit{\Delta} x = (\widehat{\mu} - \mu)x'(\mu)$ and $\mathit{\Delta} y = (\widehat{\mu} - \mu)y'(\mu)$. Hence, based on the Taylor-series affine model of $x(\mu)$ near μ, $\mathit{\Delta} x$ and $\mathit{\Delta} y$ will usually give a good approximation of the step to $\big(x(\widehat{\mu}), y(\widehat{\mu})\big)$, the next point on the trajectory.

As in the classical Newton-barrier method, primal-dual methods have a two-level structure of inner and outer iterations, with the inner iterations corresponding to the iterations of Newton's method for a given value of μ. Primal-dual methods exhibit excellent performance in the neighborhood of a trajectory. In particular, under the assumption of strict complementarity and a suitable constraint qualification, the inner iterations converge at a Q-quadratic rate; see, e.g., [26]. Moreover, the inner iterations can be terminated so that the combined sequence of inner iterates ultimately converges to x^* at a Q-superlinear rate; see, e.g., [59, 91, 93, 94].

Beyond the work associated with function evaluations, the cost of a primal-dual iteration is dominated by the cost of solving the linear system (2.52). To improve efficiency for large problems, a common approach is to use block elimination to obtain smaller "condensed" systems. As $c(x) > 0$, the $(2,2)$ block of (2.52) may be eliminated to give the following $n \times n$ system for $\mathit{\Delta} x$:

$$H_C(x,y)\mathit{\Delta} x = -\big(\nabla f(x) - J(x)^{\mathrm{T}}\pi(x,\mu)\big), \tag{2.55}$$

where the *condensed primal-dual matrix* $H_C(x,y)$ is defined as

$$H_C(x,y) = H(x,y) + J(x)^{\mathrm{T}} D(x,y)^{-1} J(x), \;\text{ with }\; D(x,y) = Y^{-1}C(x).$$

The matrix D, which is introduced for later convenience, is diagonal and positive definite, with diagonal elements $d_i = c_i/y_i$. The condensed primal-dual system can be solved by either direct or iterative methods, using (for example) an off-the-shelf Cholesky factorization or preconditioned conjugate-gradient method.

A drawback with block elimination is that significant fill-in can occur in H_C. An alternative strategy is to factorize the full $(n{+}m)\times(n{+}m)$ system in (2.52) (see, e.g., [39, 40, 48, 49]), typically after symmetrizing the system. A symmetric matrix can be created by multiplying the second block of equations in (2.52) by Y^{-1} and changing the sign of the second block of columns, giving

$$\begin{pmatrix} H & J^{\mathrm{T}} \\ J & -D \end{pmatrix} \begin{pmatrix} \mathit{\Delta} x \\ -\mathit{\Delta} y \end{pmatrix} = -\begin{pmatrix} \nabla f - J^{\mathrm{T}} y \\ D(y-\pi) \end{pmatrix}, \tag{2.56}$$

where dependencies on x, y and μ have been suppressed for brevity. As $\mu \to 0$, the diagonals of D corresponding to the active constraints grow without bound and so this particular form of symmetrization produces an increasingly ill-conditioned system. However, it can be shown that the ill-conditioning is benign as long as certain direct methods are used to factorize the matrix. (For more details, see [36, 78, 89, 90].)

In the neighborhood of a trajectory of minimizers, the primal-dual system is usually nonsingular and the iterates converge at a Q-quadratic rate. However, when the problem is nonconvex and the primal-dual iterate is far from the trajectory, there is no guarantee that a solution of the primal-dual system or condensed system exist. In this case, systems based on certain modified Hessians must be formulated.

An important component of a practical primal-dual method for nonconvex optimization is the method used to ensure convergence from any starting point. One of the most popular strategies is to require, through a line search, a sufficient decrease in a *merit function* that forces the early iterates towards the trajectory.

A number of primal-dual methods use the classical barrier function $B(x,\mu)$ as a merit function (see, e.g., [14, 63]). A different approach is based on the merit function

$$M^{\mu}(x,y) = f(x) - \mu\sum_{i=1}^{m}\ln c_i(x) - \mu\sum_{i=1}^{m}\left(\ln\left(\frac{c_i(x)y_i}{\mu}\right) + 1 - \frac{c_i(x)y_i}{\mu}\right), \tag{2.57}$$

which includes both primal and dual variables (see [34]). The function $M^{\mu}(x,y)$ is the classical barrier function $B(x,\mu)$ augmented by a weighted proximity term that measures the distance of (x,y) to the trajectory $\big(x(\mu),y(\mu)\big)$. A key property of $M^{\mu}(x,y)$ is that it is minimized with respect to both x and y at any point $(x(\mu),y(\mu))$ on the trajectory, which implies that a decrease in $M^{\mu}(x,y)$ can be used to measure the progress towards a minimizer of $B(x,\mu)$. The gradient of $M^{\mu}(x,y)$ is

$$\nabla M^{\mu}(x,y) = \begin{pmatrix} \nabla f(x) - 2\mu J(x)^{\mathrm{T}}C(x)^{-1}e - J(x)^{\mathrm{T}}y \\ C(x)e - \mu Y^{-1}e \end{pmatrix} = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}}(2\pi - y) \\ Y^{-1}C(x)\big(y-\pi\big) \end{pmatrix},$$

where $\pi = \pi(x,\mu)$ is the vector of primal multipliers (see Definition 2.2.12)). The Hessian of $M^{\mu}(x,y)$ is

$$\begin{aligned}\nabla^2 M^{\mu}(x,y) &= \begin{pmatrix} H(x,2\pi-y) + 2J(x)^{\mathrm{T}}C(x)^{-1}\mathit{\Pi} J(x) & J(x)^{\mathrm{T}} \\ J(x) & \mu Y^{-2} \end{pmatrix} \\ &= \begin{pmatrix} H\big(x, y + 2(\pi-y)\big) + 2J(x)^{\mathrm{T}}\mathit{\Pi} C(x)^{-1}J(x) & J(x)^{\mathrm{T}} \\ J(x) & D\mathit{\Pi} Y^{-1} \end{pmatrix},\end{aligned}$$

with $D = Y^{-1}C(x)$ as defined before. As $\big(x(\mu),y(\mu)\big)$ minimizes $M^{\mu}(x,y)$, it follows that $\nabla M^{\mu}(x,y) = 0$ and $\nabla^2 M^{\mu}(x,y)$ is positive semidefinite at all points on the trajectory. As a result, line-search or trust-region methods can be devised in which the local quadratic model is $\mathcal{Q}(s) = s^{\mathrm{T}}\nabla M + \frac{1}{2}s^{\mathrm{T}}H_{\mu}^{(\mathrm{pd})}s$, where

$$H_{\mu}^{(\mathrm{pd})} = \begin{pmatrix} H(x,y) + 2J(x)^{\mathrm{T}}D^{-1}J(x) & J(x)^{\mathrm{T}} \\ J(x) & D \end{pmatrix},$$

i.e., $H_{\mu}^{(\mathrm{pd})}$ is $\nabla^2 M^{\mu}(x,y)$ with π replaced by y. It can be shown that if $H_{\mu}^{(\mathrm{pd})}$ is positive definite, the solution of $H_{\mu}^{(\mathrm{pd})}s = -\nabla M$ is the unique minimizer of $\mathcal{Q}(s)$, and that $s = (\mathit{\Delta} x, \mathit{\Delta} y)$ also solves the primal-dual system (2.52). Indeed, note that a premultiplication of both sides of (2.52) by the nonsingular matrix

$$\begin{pmatrix} I & 2J(x)^{\mathrm{T}}D^{-1} \\ 0 & Y^{-1} \end{pmatrix}$$

gives the equivalent equation

$$\begin{pmatrix} H(x,y) + 2J(x)^{\mathrm{T}} D^{-1} J(x) & J(x)^{\mathrm{T}} \\ J(x) & D \end{pmatrix} \begin{pmatrix} \mathit{\Delta} x \\ \mathit{\Delta} y \end{pmatrix} = - \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}}(2\pi - y) \\ D(y - \pi) \end{pmatrix}.$$

These properties suggest a line-search method for minimizing $M^{\mu}(x,y)$ that uses the solution of the primal-dual system (2.52) as a search direction (for more details, see [34]). If $H_{\mu}^{(\mathrm{pd})}$ is sufficiently positive definite, the search direction is the unique solution of $H_{\mu}^{(\mathrm{pd})} s = -\nabla M$ (equivalent to the primal-dual system (2.52)). Otherwise, the search direction can be chosen as a the solution of a related positive-definite system $\bar{H}_{\mu}^{(\mathrm{pd})} s = -\nabla M$, with $\bar{H}_{\mu}^{(\mathrm{pd})}$ a modified Hessian that approximates $H_{\mu}^{(\mathrm{pd})}$. If the condensed matrix is formed, it can be modified "on the fly" during the factorization so that its factors are those of a positive-definite $\bar{H} + J^{\mathrm{T}} D^{-1} J$ for some implicitly defined $\bar{H}$ (see, e.g., [35]). Alternatively, the inertia-controlling LBL^{T} factorization discussed in [33, 34, 38] detects and modifies indefiniteness of the (implicitly defined) matrix $H + J^{\mathrm{T}} D^{-1} J$ while factorizing the full system (2.56). A potential drawback is that the row and column interchanges needed by the inertia-controlling factorization interfere with the row and column ordering used to maintain sparsity in the factors, producing factors that are generally less sparse than those obtained by off-the-shelf sparse-matrix software.

2.2.4 Treatment of equality constraints

Although interior methods are, strictly speaking, relevant only to inequality constraints, it is essential to take into account equality constraints as they may arise naturally as part of an optimization problem together with inequality constraints. Additionally, it is sometimes more efficient to convert an inequality constraint into an alternative form that involves one or more equality constraints. The two most common reformulations of of the single inequality constraint $c_i(x) \geq 0$ are

$$\begin{aligned} &c_i(x) - s_i = 0, \quad s_i \geq 0, \quad \text{where } s_i \text{ is called a } \textit{slack variable}\text{; or} \\ &c_i(x) + s_i \geq 0, \quad s_i = 0, \quad \text{where } s_i \text{ is called a } \textit{shift variable.} \end{aligned}$$

The slack and shift reformulations allow interior methods to be applied to inequality constraints even if no initial strictly feasible point is known, as long as suitable techniques are available for dealing with equality constraints. Moreover, within an interior method, it is extremely simple to retain strict feasibility with respect to the bound constraint $s_i \geq 0$ imposed on a slack variable, while the step to the boundary of a general nonlinear constraint must be calculated with additional evaluations of the constraint function.

Now let $\mathcal{E}$ and $\mathcal{I}$ denote a partition of the m indices of constraint functions $\{ c_i(x) \}$ such that every $i \in \mathcal{E}$ corresponds to an equality constraint $c_i(x) = 0$, and every $i \in \mathcal{I}$ corresponds to an inequality constraint $c_i(x) \geq 0$. A general nonlinearly constrained optimization problem may be written in the form

$$\begin{aligned} &\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x) \\ &\text{subject to} \;\; \begin{cases} c_i(x) = 0, & i \in \mathcal{E} \\ c_i(x) \geq 0, & i \in \mathcal{I}. \end{cases} \end{aligned} \tag{NCP}$$

The problem is to find the least value of $f(x)$ over all values of x in $\mathcal{D}$ that satisfy m_E equality constraints $c_i(x) = 0$, $i \in \mathcal{E}$, and $m_\mathcal{I}$ inequality constraints $c_i(x) \geq 0$, $i \in \mathcal{I}$. The feasible region for this problem is given by

$$\varOmega = \{ x \in \mathbb{R}^n : c_i(x) = 0, \;\; i \in \mathcal{E}, \;\; \text{and} \;\; c_i(x) \geq 0, \;\; i \in \mathcal{I} \}.$$

Let $c_\mathcal{I}(x)$ denote the subvector of components $c_i(x)$, with $i \in \mathcal{I}$, and $c_\mathcal{E}(x)$ the subvector $c_i(x)$, with $i \in \mathcal{E}$.

Definition 2.2.13 (Active, inactive, and violated constraints)**.** *For the inequality constraints* $c_\mathcal{I}(x) \geq 0$*, the* i*th constraint is said to be* active *at* $\bar{x}$ *if* $c_i(\bar{x}) = 0$*,* inactive *if* $c_i(\bar{x}) > 0$ *and* violated *if* $c_i(\bar{x}) < 0$*. For the equality constraints* $c_\mathcal{E}(x) = 0$*, the* i*th constraint is* satisfied *at* $\bar{x}$ *if* $c_i(\bar{x}) = 0$ *and* violated *at* $\bar{x}$ *if* $c_i(\bar{x}) \neq 0$*. The* active set $\mathcal{A}(\bar{x})$ *is the set of indices of the active constraints at* $\bar{x}$*, i.e.,* $\mathcal{A}(\bar{x}) = \{ i : c_i(\bar{x}) = 0 \}$*. The set of active inequality constraints at* $\bar{x}$ *is denoted by* $\mathcal{A}_\text{a}(\bar{x})$*, i.e.,* $\mathcal{A}_\text{a}(\bar{x}) = \{ i \in \mathcal{I} : c_i(\bar{x}) = 0 \}$*.*

Before discussing methods for dealing with equality constraints, the optimality conditions for the general mixed-constraint problem (NCP) will be stated. The following definition is directly analogous to the definition of a KKT point for an inequality-constrained problem (Definition 2.2.2).

Definition 2.2.14 (First-order KKT point)**.** *The* first-order KKT conditions *for problem* (NCP) *hold at the point* x^**, or, equivalently,* x^* *is a (first-order)* KKT point*, if there exists a Lagrange* *multiplier vector y^* such that*

$$\begin{aligned} c_{\mathcal{I}}(x^*) &\geq 0 \;\; \textit{and} \;\; c_{\mathcal{E}}(x^*) = 0 && \text{(feasibility)}, \\ \nabla f(x^*) &= J(x^*)^{\mathrm{T}} y^* && \text{(stationarity)}, \\ y_{\mathcal{I}}^* &\geq 0 && \text{(nonnegativity), and} \\ c_{\mathcal{I}}(x^*) \cdot y_{\mathcal{I}}^* &= 0 && \text{(complementarity)}. \end{aligned} \tag{2.58}$$

The first-order KKT conditions may be written more compactly as $F(x,y) = 0$, $c_{\mathcal{I}}(x^*) \geq 0$, $y_{\mathcal{I}} \geq 0$, with

$$F(x,y) = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y \\ c_{\mathcal{I}}(x) \cdot y_{\mathcal{I}} \\ c_{\mathcal{E}}(x) \end{pmatrix}. \tag{2.59}$$

The KKT conditions are based on the properties of constraint linearizations, and hence they are necessary conditions for optimality only when the local constraint linearizations reflect the properties of the nonlinear constraints, i.e., when a *constraint qualification* holds.

Definition 2.2.15 (LICQ for mixed constraints)**.** *The* linear independence constraint qualification *holds at the feasible point $\bar{x}$ of (NCP) if the constraint gradients $\nabla c_i(\bar{x})$, $i \in \mathcal{E} \cup \mathcal{A}_{\mathrm{a}}(\bar{x})$ are linearly independent.*

Definition 2.2.16 (MFCQ for mixed constraints)**.** *The* Mangasarian–Fromovitz constraint qualification *holds at the feasible point $\bar{x}$ of* (NCP) *if the gradients of the equality constraints at $\bar{x}$, $\nabla c_i(\bar{x})$, $i \in \mathcal{E}$, are linearly independent and if there exists a vector p such that $\nabla c_i(\bar{x})^{\mathrm{T}} p > 0$ for all $i \in \mathcal{A}_{\mathrm{a}}(\bar{x})$ and $\nabla c_i(\bar{x})^{\mathrm{T}} p = 0$ for all $i \in \mathcal{E}$.*

It must be emphasized that full row rank of $J_{\mathcal{E}}(\bar{x})$ is needed for the MFCQ to hold at $\bar{x}$. Moreover, satisfaction of the LICQ implies that the MFCQ also holds.

Theorem 2.2.7 (First-order necessary conditions for mixed constraints)**.** *If x^* is a local minimizer of problem* (NCP) *and the MFCQ holds at x^*, then x^* must be a KKT point.* □

By analogy with Definition 2.2.3 for the inequality case, the set of multipliers that satisfy the KKT conditions of Definition 2.2.14 is defined below.

Definition 2.2.17 (Acceptable Lagrange multipliers for mixed constraints)**.** *Given a KKT point x^* for problem* (NCP)*, the set of acceptable multipliers is defined as*

$$\mathcal{Y}(x^*) = \big\{\, y \in \mathbb{R}^m : \nabla f(x^*) = J(x^*)^{\mathrm{T}} y, \;\; y_{\mathcal{I}} \geq 0, \;\; \textit{and} \;\; y_{\mathcal{I}} \cdot c_{\mathcal{I}}(x^*) = 0 \,\big\}. \tag{2.60}$$

Using $\mathcal{Y}(x^*)$, second-order necessary conditions for optimality can be stated when the LICQ holds.

Theorem 2.2.8 (Second-order necessary conditions for (NCP))**.** *Suppose that x^* is a local minimizer of* (NCP) *at which the LICQ holds. Then there is a vector y^* which satisfies $y^*_{\mathcal{I}} \geq 0$, $c_{\mathcal{I}}(x^*) \cdot y^*_{\mathcal{I}} = 0$, and $\nabla f(x^*) = J(x^*)^{\mathrm{T}} y^*$, and $p^{\mathrm{T}} H(x^*, y^*) p \geq 0$ for all p satisfying $\nabla f(x^*)^{\mathrm{T}} p = 0$, $J_{\mathcal{E}}(x^*) p = 0$ and $J_{\mathrm{a}}(x^*) p \geq 0$.* □

The next theorem is analogous to Theorem 2.2.3.

Theorem 2.2.9 (Sufficient conditions for an isolated solution)**.** *The point x^* is an isolated local constrained minimizer of problem* (NCP) *if*

(**a**) *x^* is a KKT point, i.e., $c_{\mathcal{I}}(x^*) \geq 0$, $c_{\mathcal{E}}(x^*) = 0$, and there exists a nonempty set $\mathcal{Y}$ of multipliers y satisfying $y_{\mathcal{I}} \geq 0$, $c_{\mathcal{I}}(x^*) \cdot y_{\mathcal{I}} = 0$, and $\nabla f(x^*) = J(x^*)^{\mathrm{T}} y$;*

(**b**) *the MFCQ holds at x^*;*

(**c**) *for all $y \in \mathcal{Y}(x^*)$ of* (2.60) *and all nonzero p satisfying $\nabla f(x^*)^{\mathrm{T}} p = 0$, $J_{\mathcal{E}}(x^*) p = 0$ and $J_{\mathrm{a}}(x^*) p \geq 0$, there exists $\omega > 0$ such that $p^{\mathrm{T}} H(x^*, y) p \geq \omega \|p\|^2$.* □

Finally, the following theorem is analogues to Theorem 2.2.10.

Theorem 2.2.10 (Strong sufficient conditions for an isolated solution)**.** *The point x^* is an isolated local constrained minimizer of problem* (NCP) *if*

(**a**) *x^* is feasible and the LICQ holds at x^*, i.e., $\begin{pmatrix} J_{\mathcal{E}}(x^*) \\ J_{\mathrm{a}}(x^*) \end{pmatrix}$ has full row rank;*

(**b**) *x^* is a KKT point and strict complementarity holds, i.e., the (necessarily unique) multiplier y^* has the property that $[\, y^*_{\mathrm{a}} \,]_i > 0$ for all $i \in \mathcal{A}_{\mathrm{a}}(x^*)$;*

(**c**) *for all nonzero p such that $J_{\mathrm{a}}(x^*) p = 0$, there exists $\omega > 0$ such that $p^{\mathrm{T}} H(x^*, y^*) p \geq \omega \|p\|^2$.* □

Condition (**c**) of Theorem 2.2.10 is equivalent to stating that the reduced Hessian of the Lagrangian, $Z(x^*)^{\mathrm{T}} H(x^*, y^*) Z(x^*)$, is positive definite, where $Z(x^*)$ is a matrix whose columns form a basis for the null space of $\begin{pmatrix} J_{\mathcal{E}}(x^*) \\ J_{\mathrm{a}}(x^*) \end{pmatrix}$.

A classical treatment of equality constraints is to eliminate them through unconstrained minimization of a composite function that includes a *penalty* for violating $c_{\mathcal{E}}(x) = 0$—most commonly, the quadratic penalty function $\|c_{\mathcal{E}}\|_2^2/\mu$. For a general mixed-constraint problem, a *penalty-barrier* function was proposed in [31]:

$$\Phi_{\text{PB}}(x\,;\mu) = f(x) - \mu \sum_{i\in\mathcal{I}} \ln c_i(x) + \frac{1}{2\mu}\,\|c_{\mathcal{E}}\|_2^2. \tag{2.61}$$

The implicit constraints $c_{\mathcal{I}}(x) > 0$ are handled by the barrier term. Let x_μ denote an unconstrained minimizer of $\Phi_{\text{PB}}(x\,;\mu)$. A detailed analysis, analogous to the results of barrier methods, is given in [31] of the conditions under which, for sufficiently small μ, the sequence $\{\,x_\mu\,\}$ defines a differentiable *penalty-barrier trajectory* converging to x^*.

To find x_μ, stationarity of $\nabla\Phi_{\text{PB}}(x)$ must be exploited. Writing out $\nabla\Phi_{\text{PB}}(x)$ and rearranging produce a system of nonlinear equations equivalent to the condition that $\nabla\Phi_{\text{PB}}(x) = 0$:

$$F^\mu(x,y) = \begin{pmatrix} \nabla f(x) - J(x)^{\text{T}} y \\ c_{\mathcal{I}}(x)\,\cdot\,y_{\mathcal{I}} - \mu e \\ c_{\mathcal{E}}(x) + \mu y_{\mathcal{E}} \end{pmatrix} = 0, \tag{2.62}$$

where $y_{\mathcal{I}}$ and $y_{\mathcal{E}}$ represent multiplier estimates that converge to $y_{\mathcal{I}}^*$ and $y_{\mathcal{E}}^*$ as $\mu \to 0$, and, at x_μ, satisfy the relations

$$c_{\mathcal{I}}(x_\mu)\,\cdot\,y_{\mathcal{I}} = \mu e \;\text{ and }\; \mu y_{\mathcal{E}} = -c_{\mathcal{E}}(x_\mu).$$

A useful interpretation of (2.62) is that the complementarity portions of the KKT conditions (2.58) corresponding to *both* inequality and equality constraints have been perturbed.

For inequality constraints, define $\pi_{\mathcal{I}}(x,\mu)$ as

$$\pi_{\mathcal{I}}(x,\mu) = \mu \;\cdot\!/\; c_{\mathcal{I}}(x), \tag{2.63}$$

so that $\pi_{\mathcal{I}}$ is an estimate of $y_{\mathcal{I}}$ at the current iterate x for a specific value of μ; see Definition 2.2.12. To complete the definition of $\pi(x,\mu)$ for the equality constraints, define an estimate of $y_{\mathcal{E}}$ as

$$\pi_{\mathcal{E}}(x,\mu) = -c_{\mathcal{E}}(x)/\mu.$$

Application of Newton's method for equations to (2.62) gives

$$\begin{pmatrix} H & -J^{\text{T}} \\ ZJ & W \end{pmatrix} \begin{pmatrix} \mathit{\Delta} x \\ \mathit{\Delta} y \end{pmatrix} = -\begin{pmatrix} \nabla f - J^{\text{T}} y \\ W(y-\pi) \end{pmatrix}, \tag{2.64}$$

where W and Z denote diagonal matrices whose entries are

$$w_{\mathcal{I}}(x) = c_{\mathcal{I}}(x), \;\; w_{\mathcal{E}}(x) = \mu e, \;\; z_{\mathcal{I}}(y) = y_{\mathcal{I}}, \;\text{ and }\; z_{\mathcal{E}}(y) = 1. \tag{2.65}$$

The matrix in (2.64) can be symmetrized into the form (2.56), where $D = Z^{-1}W$; thus (2.64) can be solved similarly as solving the primal-dual equations (2.52).

Treating equalities via a quadratic penalty function tends to regularize the problem in the sense that, as long as μ is nonzero, the matrix $(ZJ \;\; W)$ may have full row rank even if the Jacobian $J_{\mathcal{E}}$ is rank-deficient. Consequently, one needs to modify only H to make the matrix in (2.64) nonsingular [34].

A Newton-based line-search or trust-region method can be used to solve the nonlinear equations (2.62). In the context of a line-search method, Φ_{PB} itself can be used as a merit function. Alternatively, the merit function $M^{\mu}(x,y)$ of (2.57) can be generalized by adding

$$\frac{1}{2\mu}\|c_{\mathcal{E}}(x)\|_2^2 + \frac{1}{2\mu}\|c_{\mathcal{E}}(x) + \mu y_{\mathcal{E}}\|_2^2, \tag{2.66}$$

which represents a combination of the original quadratic penalty term from (2.61) and a term that reflects proximity to the condition $\mu y_{\mathcal{E}} = -c_{\mathcal{E}}(x_\mu)$ that holds along the penalty-barrier trajectory. If x_μ is a point on the penalty-barrier trajectory and y_μ is the associated multiplier defined by (2.62), then (x_μ, y_μ) is an unconstrained minimizer of $M^{\mu}(x,y)$ for sufficiently small μ. See [34, 43] for further details.

Other than the classical penalty-barrier approach, the mixed-constraint problem (NCP) can also be solved by applying a sequential quadratic programming (SQP) method to solve a sequence of equality-constrained subproblems of the form

$$\begin{aligned} &\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) - \mu \sum_{i\in\mathcal{I}} \ln c_i(x) \\ &\text{subject to} \;\; c_i(x) = 0, \;\; i \in \mathcal{E}, \end{aligned} \tag{2.67}$$

with the value of μ converging to zero (see, e.g., [12, 22, 42, 76, 92]).

Chapter 3

Projected-Search Methods for Bound-Constrained Optimization

3.1 Introduction

This chapter describes two new classes of projected-search methods for bound-constrained problems of the form

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\;\; f(x) \quad \text{subject to} \;\; x \in \mathit{\Omega}, \tag{BC}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is a twice-continuously differentiable function and $\mathit{\Omega} = \{\, x \in \mathbb{R}^n : \ell \le x \le u \,\}$ for vectors of lower and upper bounds such that $\ell \le u$ (with all inequalities defined componentwise). The first-order optimality conditions for problem (BC) at $x^* \in \mathit{\Omega}$ are

$$x^* \in \mathit{\Omega}, \;\; \text{with} \;\; \nabla_i f(x^*) \begin{cases} \ge 0 & \text{if } x_i^* = \ell_i, \\ = 0 & \text{if } \ell_i < x_i^* < u_i, \\ \le 0 & \text{if } x_i^* = u_i, \end{cases}$$

where $\nabla_i f(x)$ denotes the i-th component of the gradient of f. These conditions impose sign conditions on the gradient at components of x^* associated with the active set $\mathcal{A}(x^*)$.

A conventional projected-search method for problem (BC) generates a sequence of feasible iterates $\{\, x_k \,\}_{k=0}^{\infty}$ such that $x_{k+1} = \mathbf{proj}_{\Omega}(x_k + \alpha_k p_k)$, where p_k is a descent direction for f at x_k, α_k is a scalar step length, and $\mathbf{proj}_{\Omega}(x)$ is the projection of x onto the feasible region, i.e.,

$$[\,\mathbf{proj}_{\Omega}(x)\,]_i = \begin{cases} \ell_i & \text{if } \; x_i < \ell_i, \\ u_i & \text{if } \; x_i > u_i, \\ x_i & \text{otherwise.} \end{cases}$$

The new iterate may be written as $x_{k+1} = x_k(\alpha_k)$, where $x_k(\alpha)$ denotes the vector $x_k(\alpha) = \mathbf{proj}_{\Omega}(x_k + \alpha p_k)$. The function $x_k(\alpha)$ defines a linear piecewise-continuous path, and the function $f\big(x_k(\alpha)\big)$ is not necessarily differentiable along $x_k(\alpha)$. In particular, $f\big(x_k(\alpha)\big)$ may have a "kink" at any $\alpha > 0$ at which $[\,p_k\,]_i \neq 0$ and either $[\,x_k + \alpha p_k\,]_i = \ell_i$ or $[\,x_k + \alpha p_k\,]_i = u_i$. This implies that it is not possible to use a line search based on the conventional Wolfe conditions. Thus, existing projected-search methods are restricted to using a search based on satisfying an Armijo-like condition along the path $x_k(\alpha)$. A commonly used Armijo-like condition is the quasi-Armijo condition:

$$f\big(x_k(\alpha_k)\big) \le f(x_k) + \alpha_k \eta_{\scriptscriptstyle A} \nabla f(x_k)^{\mathrm{T}} p_k. \tag{3.1}$$

A step that satisfies this condition is called a *quasi-Armijo* step. If γ and σ denote fixed parameters such that $\gamma > 0$ and $\sigma \in (0,1)$, then a quasi-Armijo step has the form $\alpha_k = \gamma \sigma^{t_k}$, where t_k is the smallest nonnegative integer such that the quasi-Armijo condition (3.1) is satisfied.

In this chapter, a new *quasi-Wolfe* line search is formulated that extends the benefits of a Wolfe line search to projected-search methods. The behavior of the search is similar to that of a conventional Wolfe line search, except that a step is accepted under a wider range of conditions that take into account points at which f is not differentiable. As in the unconstrained case, the quasi-Wolfe step can be computed using safeguarded polynomial interpolation and the accuracy of the step can be adjusted.

Two general classes of projected-search methods that use the new quasi-Wolfe search are proposed for solving problem (BC). These methods may be broadly categorized as being active-set methods or interior methods.

Projected-search active-set methods. The class of projected-search active-set methods is characterized by the use of a descent direction d_k computed with respect to a perturbed or extended active set (a similar set is used by Bertsekas [6]). The vector d_k may be computed in many ways, e.g., using an exact or modified Newton method or a quasi-Newton method. This direction is used as the basis for the computation of a search direction p_k, and an associated step length α_k such that

$f\big(\mathbf{proj}_{\Omega}(x_k + \alpha_k p_k)\big) < f(x_k)$. The convergence properties are established under assumptions that are typical in the analysis of projected-search methods. Moreover, it is shown that if the iterates converge to a nondegenerate stationary point, then the optimal active set is identified in a finite number of iterations. It follows that once the optimal active set has been identified, any method in this class will have the same convergence rate as its unconstrained counterpart.

In addition, a new active-set method, a quasi-Newton projected-search method UBOPT is proposed as an extension of the limited-memory reduced-Hessian method L-RHR of Leonard [66] and Gill and Leonard [47]. The method is based on the work of Fenelon [27] and Siegel [81], who independently proposed methods that exploit the fact that quasi-Newton methods accumulate approximate curvature in a sequence of expanding subspaces. In particular, Fenelon considered a method in which the search direction is computed using a reduced matrix that represents the approximate Hessian in the subspace. Though the subspace and this reduced matrix increase in dimension at each iteration, the dimension is limited to some fixed number and only the most recent information is used to define the subspace and the matrix.

Projected-search interior methods. The class of projected-search interior methods combines a traditional interior method with a projected-search algorithm for the minimization of a sequence of merit functions parameterized by a positive scalar. The underlying Newton or approximate Newton directions are projected onto a subset of the feasible region defined by perturbing the bounds. Global convergence properties that are analogous to those of the active-set methods are established.

The rest of the chapter is organized as follows. Section 3.2 defines a new step type, a *quasi-Wolfe step*, and establishes theoretical results for the implementation of the quasi-Wolfe search. Section 3.3 describes a class of projected-search active-set methods that utilizes the quasi-Wolfe search, and establishes the convergence results. Section 3.4 begins with a brief review of the L-RHR method of Leonard [66] and Gill and Leonard [47] for unconstrained optimization, and then introduces the new method UBOPT for unconstrained and bound-constrained optimization that is formulated based on the framework of the projected-search active-set methods, Section 3.5 proposes a class of projected-search interior methods and establish its convergence results.

3.2 The Quasi-Wolfe Search

3.2.1 The quasi-Wolfe step

At each iteration k, projected-search methods perform a search on the univariate function

$$\psi_k(\alpha) = f\big(x_k(\alpha)\big) = f\big(\mathbf{proj}_{\Omega}(x_k + \alpha p_k)\big),$$

instead of $\phi_k(\alpha) = f(x_k + \alpha p_k)$. It is a substantially more difficult task because ψ_k is only piecewise continuously differentiable, with a finite number of jump discontinuities in the derivative (see Section 3.2.2 below). In the following discussion, the suffix k is omitted if the iteration index is not relevant to the discussion.

The definition of a quasi-Wolfe step involves the left and right derivatives $\psi'_-(\alpha)$ and $\psi'_+(\alpha)$ of ψ at α, which are defined as

$$\psi'_-(\alpha) = \lim_{\beta \to \alpha^-} \psi'(\beta) \quad \text{and} \quad \psi'_+(\alpha) = \lim_{\beta \to \alpha^+} \psi'(\beta).$$

Definition 3.2.1. *Let η_A and η_W be constant scalars such that $0 < \eta_A < \eta_W < 1$. A step $\alpha > 0$ is called a* quasi-Wolfe step *if it satisfies the quasi-Armijo condition*

($\mathbf{C}_1$) $\psi(\alpha) \le \psi(0) + \alpha \eta_A \psi'_+(0)$,

and at least one of the following conditions:

($\mathbf{C}_2$) $|\psi'_-(\alpha)| \le \eta_W |\psi'_+(0)|$;

($\mathbf{C}_3$) $|\psi'_+(\alpha)| \le \eta_W |\psi'_+(0)|$;

($\mathbf{C}_4$) *ψ is not differentiable at α and $\psi'_-(\alpha) \le 0 \le \psi'_+(\alpha)$.*

Figure 3.1 depicts three examples of a kink point satisfying the quasi-Wolfe conditions.

The properties of the quasi-Wolfe search are characterized by extending the framework for the differentiable case. In particular, the discussion makes extensive use of the auxiliary function

$$\omega(\alpha) = \psi(\alpha) - \big(\psi(0) + \alpha \eta_A \psi'_+(0)\big), \quad \text{with} \quad \omega'_\pm(\alpha) = \psi'_\pm(\alpha) - \eta_A \psi'_+(0). \tag{3.2}$$

The following lemma is used to establish the propositions below.

Lemma 3.2.1. *Let a, $b \in \mathbb{R}$ be such that $0 \le a < b$, and assume that θ is a univariate, continuous, piecewise continuously differentiable function with a finite number of jump discontinuities in the derivative.*

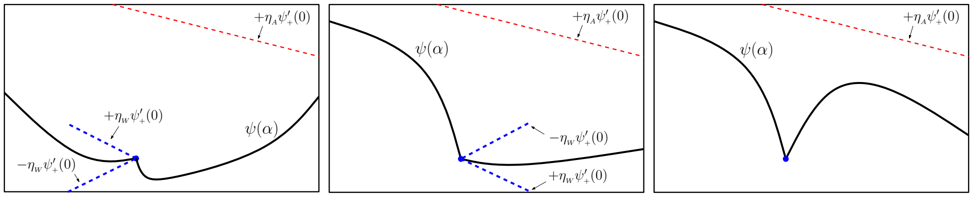

Figure 3.1: Three examples of a kink point satisfying the quasi-Wolfe conditions. The left, center and right figures depict kink points satisfying conditions $(\mathbf{C}_2)$, $(\mathbf{C}_3)$ and $(\mathbf{C}_4)$ respectively. The slope of each dashed line is marked.

(**a**) *If $\theta'_+(a) \le 0$ and $\theta(a) \le \theta(b)$, then there exists an $x \in (a,b)$ such that*

$$\theta'_-(x) \le 0 \le \theta'_+(x).$$

(**b**) *If $\theta'_+(a) < 0$ and $\theta'_-(b) > 0$ then there exists an $x \in (a,b)$ such that*

$$\theta'_-(x) \le 0 \le \theta'_+(x).$$

If θ is differentiable at x then the inequalities in the conclusions of parts (**a**) *and* (**b**) *hold as equalities.*

Proof. For part (**a**), let $a = s_0 < s_1 < s_2 < \cdots < s_t < s_{t+1} = b$, where s_1, s_2, $\ldots$, s_t represent all the points in (a,b) at which θ is nondifferentiable. First, suppose that $\theta'_+(y) \le 0$ for all $y \in (a,b)$. Then θ is continuously differentiable and nonincreasing within each subinterval $[s_j, s_{j+1}]$ for $j = 0$, 1, $\ldots$, t. It follows that $\theta(a) \ge \theta(s_1) \ge \cdots \ge \theta(s_t) \ge \theta(b)$. By assumption, this is true only when $\theta(a) = \theta(b)$, which implies that $\theta(a) = \theta(s_1)$. Thus, by Rolle's Theorem, there exists an $x \in (a, s_1) \subset [a,b]$ such that $\theta'(x) = \theta'_\pm(x) = 0$. Now suppose there is a $y \in (a,b)$ such that $\theta'_+(y) > 0$, and let $x = \inf\big\{\, y \in (a,b) : \theta'_+(y) > 0 \,\big\}$. Then $x \in (a,b)$, $\theta'_+(x) \ge 0$, and $\theta'_-(x) = \lim_{y \to x^-} \theta'_+(y) \le 0$. For part (**b**), let $x = \inf\big\{\, y \in (a,b) : \theta'_+(y) > 0 \,\big\}$. Then $x \in (a,b)$, $\theta'_+(x) \ge 0$, and $\theta'_-(x) = \lim_{y \to x^-} \theta'_+(x) \le 0$. □

The next result establishes conditions on f and Ω that guarantee the existence of a quasi-Wolfe step at each iteration.

Proposition 3.2.1. *Let f be a scalar-valued continuously differentiable function defined on $\Omega = \big\{\, x \in \mathbb{R}^n : \ell \le x \le u \,\big\}$. Assume that $x_0 \in \Omega$ is chosen such that the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded, and assume that $\big\{\, p_k \,\big\}$ is a sequence of descent directions. If η_A and η_W are fixed scalars such* *that* $0 < \eta_{\scriptscriptstyle A} \le \eta_{\scriptscriptstyle W} < 1$*, then at every iteration* k *either there exists an* $\alpha_{\scriptscriptstyle L}^{(k)} > 0$ *and an interval* $(\alpha_{\scriptscriptstyle L}^{(k)}, \alpha_{\scriptscriptstyle U}^{(k)})$ *such that every* $\alpha \in (\alpha_{\scriptscriptstyle L}^{(k)}, \alpha_{\scriptscriptstyle U}^{(k)})$ *is a quasi-Wolfe step, or there exists a quasi-Wolfe step that satisfies the condition* $(\mathbf{C}_4)$*.*

Proof. We omit the suffix k and write $\psi(\alpha) = f\big(\mathbf{proj}_{\,\Omega}(x + \alpha p)\big)$. First, it will be shown that there exists a positive scalar σ such that the function ω of (3.2) satisfies $\omega(\alpha) < 0$ for all $\alpha \in (0, \sigma)$. As $\psi'_+(0) = \nabla f(x)^{\mathrm{T}} p < 0$ and $\eta_{\scriptscriptstyle A} < 1$, it must hold that

$$\omega'_+(0) = (1 - \eta_{\scriptscriptstyle A})\psi'_+(0) < 0,$$

in which case there must be a scalar σ ($\sigma > 0$) such that $\omega(\alpha) < 0$ for all $\alpha \in (0, \sigma)$. It follows that there exists a $\sigma_1 \in (0, \sigma)$ such that $\omega(\sigma_1) < 0$.

From the compactness of the level set $\mathcal{L}\big(f(x_0)\big)$, $\psi(\alpha)$ is bounded below by some constant ψ_{low}, i.e., $\psi(\alpha) \ge \psi_{\text{low}}$ for all $\alpha \in [0, \infty)$. As $\psi(0) + \alpha\eta_{\scriptscriptstyle A}\psi'_+(0) \to -\infty$ as $\alpha \to \infty$, there must exist a positive σ_2 such that $\psi(0) + \sigma_2\eta_{\scriptscriptstyle A}\psi'_+(0) = \psi_{\text{low}}$, and we have

$$\omega(\sigma_2) = \psi(\sigma_2) - \psi(0) - \sigma_2\eta_{\scriptscriptstyle A}\psi'_+(0) = \psi(\sigma_2) - \psi_{\text{low}} \ge 0.$$

Given scalars σ_1 and σ_2 ($0 \le \sigma_1 < \sigma_2$) such that $\omega(\sigma_1) < 0$ and $\omega(\sigma_2) \ge 0$, the intermediate-value theorem states that there must exist at least one positive α such that $\omega(\alpha) = 0$. Let β denote the least positive root of $\omega(\alpha) = 0$, then $\omega(\alpha) < 0$ for all $\alpha \in (0, \beta)$. As $\omega(0) = 0$, $\omega(\beta) = 0$, and $\omega'_+(0) < 0$, by Lemma 3.2.1 ((**a**)), there exists an $\xi \in (0, \beta)$ such that

$$\omega'_-(\xi) \le 0 \le \omega'_+(\xi), \ \text{ or, equivalently, } \ \psi'_-(\xi) \le \eta_{\scriptscriptstyle A}\psi'_+(0) \le \psi'_+(\xi).$$

By construction, $\xi \in (0, \beta)$, which implies that $\omega(\xi) \le 0$, or equivalently, ξ satisfies the quasi-Armijo condition $(\mathbf{C}_1)$. If $\psi'_+(\xi) \le 0$, then the inequality $\eta_{\scriptscriptstyle A} < \eta_{\scriptscriptstyle W}$ implies that ξ is a quasi-Wolfe step that satisfies the derivative condition $(\mathbf{C}_3)$. By the piecewise continuity of $\psi'_+(\alpha)$, there exists an $\alpha_{\scriptscriptstyle L} > 0$ and an interval $(\alpha_{\scriptscriptstyle L}, \alpha_{\scriptscriptstyle U})$ such that every $\alpha \in (\alpha_{\scriptscriptstyle L}, \alpha_{\scriptscriptstyle U})$ is a quasi-Wolfe step. Otherwise, if $\psi'_+(\xi) > 0$, then ξ is a quasi-Wolfe step that satisfies the condition $(\mathbf{C}_4)$. □

The following result is analogous to Proposition 2.1.3 and motivates the first stage of a quasi-Wolfe search.

Proposition 3.2.2. *Let* $\{\, \alpha_i \,\}_{i=0}^{\infty}$ *be a strictly monotonically increasing sequence with* $\alpha_0 = 0$*. Let* ψ *be a continuous piecewise-differentiable univariate function whose derivative has a finite number of jump discontinuities. Assume that* $\psi'_+(0) < 0$ *and define* $\omega(\alpha) = \psi(\alpha) - \big(\psi(0) + \alpha\eta_{\scriptscriptstyle A}\psi'_+(0)\big)$ *with*

$0 < \eta_A < 1$. *If there exists a least bounded index j such that at least one of the following "stage-one" conditions is true:*

(**a**) *α_j is a quasi-Wolfe step;*

(**b**) *$\omega(\alpha_j) \geq \omega(\alpha_{j-1})$; or*

(**c**) *$\omega'_-(\alpha_j) \geq 0$,*

then there exists a quasi-Wolfe step $\alpha^* \in [\alpha_{j-1}, \alpha_j]$.

Proof. Observe that α_{j-1} must satisfy none of the conditions (**a**)–(**c**), otherwise j would not be the least index. This implies that $\omega(\alpha_{j-1}) < \omega(\alpha_{j-2}) < \cdots < \omega(\alpha_0) = 0$ from (**b**), and $\omega'_-(\alpha_{j-1}) < 0$ from (**c**).

The first step is to show that

$$\omega'_+(\alpha_{j-1}) < 0. \tag{3.3}$$

If $\omega'(\alpha_{j-1})$ exists, then $\omega'_+(\alpha_{j-1}) = \omega'_-(\alpha_{j-1}) < 0$. If $\omega'(\alpha_{j-1})$ does not exist, then ((**c**)) implies that $\omega'_-(\alpha_{j-1}) = \psi'_-(\alpha_{j-1}) - \eta_A\psi'_+(0) < 0$, in which case $\psi'_-(\alpha_{j-1}) < 0$ because $\psi'_+(0) < 0$ by assumption. As ($\mathbf{C}_4$) cannot hold at α_{j-1}, it follows that $\psi'_+(\alpha_{j-1}) < 0$. Now, if ($\mathbf{C}_3$) does not hold at α_{j-1} then $\psi'_+(\alpha_{j-1}) < \eta_W\psi'_+(0) < \eta_A\psi'_+(0)$. Thus, $\omega'_+(\alpha_{j-1}) = \psi'_+(\alpha_{j-1}) - \eta_A\psi'_+(0) < 0$. The inequality (3.3) is used in the proofs that follow.

Case 1. If (**a**) is true, the proposition holds trivially.

Case 2. If (**b**) is true, let $\bar{\alpha} = \sup\big\{\, \alpha \in [\alpha_{j-1}, \alpha_j] : \omega(\beta) \leq 0 \text{ for all } \beta \in [\alpha_{j-1}, \alpha] \,\big\}$. If $\bar{\alpha} = \alpha_j$, then $\omega(\bar{\alpha}) = \omega(\alpha_j) \geq \omega(\alpha_{j-1})$; if $\bar{\alpha} < \alpha_j$, then by the continuity of ω, $\omega(\bar{\alpha}) = 0 > \omega(\alpha_{j-1})$. In either case, as $\omega'_+(\alpha_{j-1}) < 0$ by (3.3), part (**a**) of Lemma 3.2.1 implies that there exists an $\alpha^* \in [\alpha_{j-1}, \bar{\alpha}]$ such that

$$\omega'_-(\alpha^*) \leq 0 \leq \omega'_+(\alpha^*).$$

This implies that

$$\psi'_-(\alpha^*) \leq \eta_A\psi'_+(0) \leq \psi'_+(\alpha^*).$$

From the definition of $\bar{\alpha}$, α^* satisfies the quasi-Armijo condition ($\mathbf{C}_1$). As $\psi'_-(\alpha^*) < 0$, if $\psi'_+(\alpha^*) \geq 0$, then α^* is a quasi-Wolfe step by ($\mathbf{C}_4$). Alternatively, if $\psi'_+(\alpha^*) < 0$, then

$$\eta_W\psi'_+(0) < \eta_A\psi'_+(0) \leq \psi'_+(\alpha^*) < 0,$$

and again, α^* is a quasi-Wolfe step by ($\mathbf{C}_3$).

Case 3. Finally, consider the case where (**c**) is true, i.e., $\omega'_-(\alpha_j) \geq 0$. By (3.3), $\omega'_+(\alpha_{j-1}) < 0$. If $\omega(\alpha) \leq 0$ for all $\alpha \in [\alpha_{j-1}, \alpha_j]$, then either $\omega'_-(\alpha_j) = 0$ such that α_j is a quasi-Wolfe step, or part (**b**) of Lemma 3.2.1 establishes the existence of a step $\alpha^* \in (\alpha_{j-1}, \alpha_j)$ such that

$$\omega'_-(\alpha^*) \leq 0 \leq \omega'_+(\alpha^*),$$

and α^* satisfies the quasi-Armijo condition $(\mathbf{C}_1)$. Otherwise, let $\bar{\alpha} = \sup\big\{\, \alpha \in [\alpha_{j-1}, \alpha_j] : \omega(\beta) \leq 0$ for all $\beta \in [\alpha_{j-1}, \alpha] \,\big\}$. By the continuity of ω, $\omega(\bar{\alpha}) = 0 > \omega(\alpha_{j-1})$. It follows from part (**a**) of Lemma 3.2.1 that there exists a step $\alpha^* \in [\alpha_{j-1}, \bar{\alpha}]$ such that

$$\omega'_-(\alpha^*) \leq 0 \leq \omega'_+(\alpha^*),$$

and α^* satisfies the quasi-Armijo condition $(\mathbf{C}_1)$. The same argument used for the preceding case shows that α^* is a quasi-Wolfe step. □

The second stage of a quasi-Wolfe search is based on the following proposition.

Proposition 3.2.3. *Let ψ and ω be defined as in* Proposition 3.2.2. *Assume there exist distinct points α_{low} and α_{high} such that*

(**a**) $\omega(\alpha_{\text{low}}) \leq 0$;

(**b**) $\omega(\alpha_{\text{low}}) \leq \omega(\alpha_{\text{high}})$; *and*

(**c**) $\omega'_+(\alpha_{\text{low}}) < 0$ *if* $\alpha_{\text{low}} < \alpha_{\text{high}}$ *or* $\omega'_-(\alpha_{\text{low}}) > 0$ *if* $\alpha_{\text{low}} > \alpha_{\text{high}}$,

then there exists a quasi-Wolfe step $\alpha^* \in \mathcal{I}$, where $\mathcal{I}$ is the interval defined with endpoints α_{low} and α_{high}.

Proof. First, consider the case where $\alpha_{\text{low}} < \alpha_{\text{high}}$. Let $\bar{\alpha} = \sup\big\{\, \alpha \in \mathcal{I} : \omega(\beta) \leq 0 \,\big\}$ for all $\beta \in [\alpha_{\text{low}}, \alpha]$. By the continuity of ω, $\omega(\bar{\alpha}) = 0 \geq \omega(\alpha_{\text{low}})$. It follows from part (**a**) of Lemma 3.2.1 that there exists a step $\alpha^* \in [\alpha_{\text{low}}, \bar{\alpha}]$ such that $\omega(\alpha^*) \leq 0$ and

$$\omega'_-(\alpha^*) \leq 0 \leq \omega'_+(\alpha^*).$$

The same argument used in Proposition 3.2.2 shows that α^* is a quasi-Wolfe step.

For the case $\alpha_{\text{low}} > \alpha_{\text{high}}$, let $\widetilde{\omega}(\alpha) = \omega(\alpha_{\text{low}} + \alpha_{\text{high}} - \alpha)$. Then $\widetilde{\omega}(\alpha_{\text{high}}) = \omega(\alpha_{\text{low}}) \leq 0$, and $\widetilde{\omega}'_+(\alpha_{\text{high}}) = -\omega'_-(\alpha_{\text{low}}) < 0$. Let $\bar{\alpha} = \sup\big\{\, \alpha \in \mathcal{I} : \widetilde{\omega}(\beta) \leq 0 \,\big\}$ for all $\beta \in [\alpha_{\text{high}}, \alpha]$. The continuity of $\widetilde{\omega}$ implies that $\widetilde{\omega}(\bar{\alpha}) = 0 \geq \widetilde{\omega}(\alpha_{\text{high}})$. It follows from part (**a**) of Lemma 3.2.1 that there exists a step $\beta^* \in [\alpha_{\text{high}}, \bar{\alpha}]$ such that $\widetilde{\omega}(\beta^*) \leq 0$ and

$$\widetilde{\omega}'_-(\beta^*) \leq 0 \leq \widetilde{\omega}'_+(\beta^*).$$

Let $\alpha^* = \alpha_{\text{low}} + \alpha_{\text{high}} - \beta^*$, then $\alpha^* \in \mathcal{I}$, $\omega(\alpha^*) \leq 0$ and

$$\omega'_-(\alpha^*) = -\widetilde{\omega}'_+(\beta^*) \leq 0 \leq -\widetilde{\omega}'_-(\beta^*) = \omega'_+(\alpha^*).$$

It follows that α^* is a quasi-Wolfe step. □

Although the implementation of a quasi-Wolfe search is similar to that of a Wolfe line search, there are a number of crucial practical issues associated with the potential nondifferentiability of the line-search function $\psi(\alpha)$. These issues include the definition of the derivatives of the function $\psi(\alpha)$ and the computation of a new estimate of a quasi-Wolfe step.

3.2.2 Derivatives of the search function

The purpose of this section is to establish expressions for the left- and right-derivatives of the search function $\psi(\alpha) = f\big(x(\alpha)\big)$, where $x(\alpha)$ is the vector $\mathbf{proj}_{\Omega}(x + \alpha p)$ with components

$$x_i(\alpha) = \begin{cases} \ell_i & \text{if } \ x_i + \alpha p_i < \ell_i, \\ u_i & \text{if } \ x_i + \alpha p_i > u_i, \\ x_i + \alpha p_i & \text{if } \ \ell_i \leq x_i + \alpha p_i \leq u_i. \end{cases}$$

First, the derivatives of $x(\alpha)$ are considered. Under the assumptions that x is feasible and α is positive, it must hold that if $x_i + \alpha p_i < \ell_i$ then $p_i < 0$, and if $x_i + \alpha p_i > u_i$, then $p_i > 0$. This implies that the right derivative of $x(\alpha)$ with respect to α is given by

$$[x'_+(\alpha)]_i = \begin{cases} 0 & \text{if } \ x_i(\alpha) = \ell_i \ \text{ and } \ p_i < 0, \\ 0 & \text{if } \ x_i(\alpha) = u_i \ \text{ and } \ p_i > 0, \\ p_i & \text{otherwise.} \end{cases}$$

The vector $x'_+(\alpha)$ may be expressed in terms of $P_x(p)$, the *projected direction of p at x*, which is defined as

$$[\,P_x(p)\,]_i = \begin{cases} 0 & \text{if } \ x_i = \ell_i \ \text{ and } \ p_i < 0, \\ 0 & \text{if } \ x_i = u_i \ \text{ and } \ p_i > 0, \\ p_i & \text{otherwise.} \end{cases}$$

The vector $P_x(p)$ represents the projection of p onto the closure of the set of feasible directions at $x(\alpha)$. If $x(\alpha)$ is differentiable at a point α, then

$$x'(\alpha) = x'_+(\alpha) = P_{x(\alpha)}(p). \tag{3.4}$$

If $x(\alpha)$ is not differentiable at α then there must be at least one index i such that

$$(x_i + \alpha p_i = \ell_i \ \text{ and } \ p_i < 0) \quad \text{or} \quad (x_i + \alpha p_i = u_i \ \text{ and } \ p_i > 0).$$

An α satisfying one of these conditions is called a *kink step with respect to i* and it also must hold that $x'_+(\alpha) \neq x'_-(\alpha)$. In order to compute the left derivative $x'_-(\alpha)$, consider the values of $x'(\beta)$ as β approaches α from below. If α is a kink step with respect to i then $x_i + \beta p_i$ is feasible for all β sufficiently close to α and it follows from (3.4) that $x'_i(\beta) = p_i$. Combining this value with the components of $x'_i(\beta)$ associated with the differentiable case gives $x'_-(\alpha) = P^-_{x(\alpha)}(p)$, where

$$[\,P^-_{x(\alpha)}(p)\,]_i = \begin{cases} p_i & \text{if } \ \alpha \ \text{ is a kink step with respect to } \ i, \\ [\,P_{x(\alpha)}(p)\,]_i & \text{otherwise.} \end{cases}$$

The derivatives of the function $\psi(\alpha)$ can now be considered. If $\psi(\alpha)$ is differentiable at α, then the chain rule gives

$$\psi'(\alpha) = \frac{d}{d\alpha} f\big(x(\alpha)\big) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} \frac{d}{d\alpha} x(\alpha) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} x'(\alpha).$$

Using this expression with the expression (3.4) for $x'(\alpha)$ gives

$$\psi'(\alpha) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} P_{x(\alpha)}(p).$$

If $\psi(\alpha)$ is not differentiable at α, then α is a kink step and $\psi'_-(\alpha) \neq \psi'_+(\alpha)$. For any α, $\lim_{\beta\to\alpha^+} x'(\beta) = x'_+(\alpha)$, and $\lim_{\beta\to\alpha^-} x'(\beta) = x_-(\alpha)$. It follows that the right- and left-derivatives of $\psi_+(\alpha)$ with respect to α are given by

$$\psi'_+(\alpha) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} x'_+(\alpha) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} P_{x(\alpha)}(p),$$

and

$$\psi'_-(\alpha) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} x_-(\alpha) = \nabla f\big(x(\alpha)\big)^{\mathrm{T}} P^-_{x(\alpha)}(p).$$

These expressions imply that there is a jump of magnitude $\big|p_i \nabla_i f\big(x(\alpha)\big)\big|$ in the derivative of ψ at a kink step with respect to i.

3.2.3 Computing a quasi-Wolfe step

As in the Wolfe line search discussed in Section 2.1.3, a quasi-Wolfe search may be regarded as having two stages. Algorithm 2 gives a schematic outline of a quasi-Wolfe search.The first stage begins with an initial step length α_0 and continues with steps of increasing magnitude until one of three things happens: an acceptable step length is found; an interval that contains a quasi-Wolfe step is found; or the step is considered to be unbounded. In practice, the search is terminated if the computed step length exceeds a preassigned upper bound $\alpha_{\max}$ during the first-stage iterations. If the search terminates at $\alpha_{\max}$ without finding an interval containing a quasi-Wolfe step, then every step computed up to that point satisfies the quasi-Armijo condition.

If the first stage terminates with a bounded step, the second stage repeatedly calls a function `Stage_Two`(α_{low}, α_{high}), where

(**a**) the interval bounded by α_{low} and α_{high} contains a quasi-Wolfe step;

(**b**) among all the step lengths generated so far, α_{low} gives the least value of ω;

(**c**) α_{high} is chosen so that $\omega'_+(\alpha_{\text{low}}) < 0$ if $\alpha_{\text{low}} < \alpha_{\text{high}}$, or $\omega'_-(\alpha_{\text{low}}) > 0$ if $\alpha_{\text{low}} > \alpha_{\text{high}}$.

A major difference between a Wolfe and a quasi-Wolfe search concerns how interpolation is used to find new steps in the second stage. Each time `Stage_Two`(α_{low}, α_{high}) is invoked, a new trial step α_{new} is generated. In the differentiable case, α_{new} is usually obtained by polynomial interpolation using the value of ϕ and its derivatives at α_{low} and α_{high}. If the univariate search function is only piecewise differentiable, there may be kink points between α_{low} and α_{high}, in which case a conventional interpolation approach may not provide a good estimate of a quasi-Wolfe step. One strategy to speed convergence in this situation is to search for the kink step (if it exists) between α_{low} and α_{high} that is closest to α_{low}. This approach is justified by the following argument. If a new point α_{new} is not a quasi-Wolfe step, then based on Proposition 3.2.3, the end points α_{low} and α_{high} are updated to α_{low} and α_{new} in two cases:

Algorithm 2 Schematic outline of a quasi-Wolfe search.

```
function QUASI_WOLFE_SEARCH($\alpha$)
    restriction: $\alpha > 0$;
    constants: $\eta_A \in (0, \frac{1}{2})$, $\eta_W \in (\eta_A, 1)$, $\gamma_e > 1$, $\alpha_{\max} \in (0, +\infty)$;
    $\alpha \leftarrow \min\{\alpha, \alpha_{\max}\}$;   $\alpha_{\text{old}} \leftarrow 0$;
    while $\alpha$ is not a quasi-Wolfe step and $\alpha \neq \alpha_{\max}$ do
        if $\omega(\alpha) \geq \omega(\alpha_{\text{old}})$ then
            $\alpha \leftarrow$ Stage_Two($\alpha_{\text{old}}, \alpha$); break;
        else if $\omega'_{-}(\alpha) \geq 0$ then
            $\alpha \leftarrow$ Stage_Two($\alpha, \alpha_{\text{old}}$); break;
        else
            $\alpha_{\text{old}} \leftarrow \alpha$;   $\alpha \leftarrow \min\{\gamma_e \alpha, \alpha_{\max}\}$;          [Increase $\alpha$ towards $\alpha_{\max}$]
        end if
    end while
    return $\alpha$;
end function
function STAGE_TWO($\alpha_{\text{low}}, \alpha_{\text{high}}$)
    restriction: $\omega(\alpha_{\text{low}}) \leq \omega(\alpha_{\text{high}})$;
    Choose $\alpha_{\text{new}}$ in the interior of the interval defined by $\alpha_{\text{low}}$ and $\alpha_{\text{high}}$;
    if $\alpha_{\text{new}}$ is a quasi-Wolfe step then
        return $\alpha_{\text{new}}$;
    else if $\omega(\alpha_{\text{new}}) \geq \omega(\alpha_{\text{low}})$ then
        return Stage_Two($\alpha_{\text{low}}, \alpha_{\text{new}}$);
    else if $\omega'_{+}(\alpha_{\text{new}}) < 0$ and $\alpha_{\text{low}} < \alpha_{\text{high}}$ then
        return Stage_Two($\alpha_{\text{new}}, \alpha_{\text{high}}$);
    else if $\omega'_{-}(\alpha_{\text{new}}) > 0$ and $\alpha_{\text{low}} > \alpha_{\text{high}}$ then
        return Stage_Two($\alpha_{\text{new}}, \alpha_{\text{high}}$);
    else
        return Stage_Two($\alpha_{\text{new}}, \alpha_{\text{low}}$);
    end if
end function
```

Case (1). $\omega(\alpha_{\text{new}}) \geq \omega(\alpha_{\text{low}})$;

Case (2). $\omega'_+(\alpha_{\text{new}}) < 0$ if $\alpha_{\text{high}} < \alpha_{\text{low}}$, or $\omega'_-(\alpha_{\text{new}}) > 0$ if $\alpha_{\text{high}} > \alpha_{\text{low}}$.

In these cases, the new interval bounded by α_{low} and α_{new} will not contain a kink step. In the remaining case:

Case (3). $\omega'_+(\alpha_{\text{new}}) \geq 0$ if $\alpha_{\text{high}} < \alpha_{\text{low}}$, or $\omega'_-(\alpha_{\text{new}}) \leq 0$ if $\alpha_{\text{high}} > \alpha_{\text{low}}$,

the new interval will be bounded by α_{high} and α_{new}, but may contain kink points. However, the new interval must contain at least one fewer kink point.

The search for the kink points proceeds as follows. At the first time invoking the function `Stage_Two`(α_{low}, α_{high}), the kink steps are computed in $O(n)$ floating-point operations (flops) from

$$\kappa_i = \begin{cases} (u_i - x_i)/p_i & \text{if } \; p_i > 0, \\ (\ell_i - x_i)/p_i & \text{if } \; p_i < 0, \\ \infty & \text{if } \; p_i = 0. \end{cases}$$

As the interval bounded by α_{low} and α_{high} contains a quasi-Wolfe step, only the kink steps within that interval need be stored. These steps are then sorted in decreasing order within $O(n \log n)$ flops using a heapsort algorithm (see, e.g., Williams [85], Knuth [65, Section 5.2.3]). The kink step closest to α_{low}, say κ_1^*, is either the smallest or the largest kink step within the interval of uncertainty, depending on whether α_{low} is smaller or greater than α_{high}. Once κ_1^* has been found, the search for κ_l^* $(l > 1)$ is made towards α_{low} starting at the kink step κ_{l-1}^* from the preceding iteration. To prevent the iterations from lingering at **Case (3)** for too long, an upper limit is imposed on the number of consecutive kink steps as trial steps. If this limit is reached, a new trial step is generated by bisection.

Once all the kinks in the interval of uncertainty have been eliminated, conventional polynomial interpolation may be used to generate a new step length. However, some care is necessary to choose the appropriate left or right derivative for use in the interpolation (see Section 3.2.2).

In the following, two types of projected-search methods are formulated that utilize the quasi-Wolfe line search. These methods may be broadly categorized as active-set methods and interior methods.

3.3 Projected-Search Active-Set Methods

3.3.1 The general framework

Given an initial $x_0 \in \mathit{\Omega}$, consider the sequence of iterates $\{\, x_k \,\}$ that satisfies $x_{k+1} = x_k(\alpha_k) = \mathbf{proj}_{\mathit{\Omega}}(x_k + \alpha_k p_k)$, where α_k is a quasi-Wolfe step, and p_k is a descent direction for f at x_k. The search direction p_k is based on the components of a feasible descent direction d_k computed in terms of a *working set* of indices at x_k such that

$$\begin{aligned} \mathcal{W}_k = \{\, i : [x_k]_i \le \ell_i + \epsilon_k \ \text{ and } \ \nabla_i f(x_k) > 0 \ \text{ or} \\ [x_k]_i \ge u_i - \epsilon_k \ \text{ and } \ \nabla_i f(x_k) < 0 \,\}, \end{aligned} \tag{3.5}$$

where ϵ_0 and ϵ are fixed positive parameters, and $\epsilon_k = \min\{\epsilon, \|\mathit{\Pi}_{k-1}^{\mathrm{T}} \nabla f(x_{k-1})\|\}$ for $k \ge 1$, with $\mathit{\Pi}_{k-1}$ the matrix of columns of the identity matrix of order n associated with the indices in the complement of $\mathcal{W}_{k-1}$ in $\{\, 1, 2, \ldots, n \,\}$. The matrix $\mathit{\Pi}_{k-1}\mathit{\Pi}_{k-1}^{\mathrm{T}}$ represents the projection $P_{\mathcal{W}_{k-1}}$ with respect to the set $\mathcal{W}_{k-1}$, i.e., for any $d \in \mathbb{R}^n$ it holds that $\mathit{\Pi}_{k-1}\mathit{\Pi}_{k-1}^{\mathrm{T}} d = P_{\mathcal{W}_{k-1}}(d)$, with

$$[\, P_{\mathcal{W}_{k-1}}(d) \,]_i = \begin{cases} 0 & \text{if } \ i \in \mathcal{W}_{k-1}, \\ d_i & \text{if } \ i \notin \mathcal{W}_{k-1}. \end{cases}$$

The search direction p_k is defined in terms of any direction d_k such that $d_k = \mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} d_k$, and $\nabla f(x_k)^{\mathrm{T}} d_k < 0$. Once d_k is determined, the components of d_k are modified if necessary to give a search direction p_k such that $[\, p_k \,]_i = \max\{\, [\, d_k \,]_i, 0 \,\}$ if $[\, x_k \,]_i \le \ell_i + \epsilon_k$ and $[\, p_k \,]_i = \min\{\, [\, d_k \,]_i, 0 \,\}$ if $[\, x_k \,]_i \ge u_i - \epsilon_k$. This additional step guarantees convergence in the situation where iterates approach a boundary point from the interior of the feasible region—a phenomenon known as zigzagging or jamming (see Bertsekas [5]). The vector p_k satisfies $p_k = \mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} p_k$, and retains the descent property of d_k. For example, if $[\, d_k \,]_i \ne 0$ and $[\, x_k \,]_i \le \ell_i + \epsilon_k$, then the definition of $\mathcal{W}_k$ implies that $\nabla_i f(x_k) \le 0$. If $[\, p_k \,]_i > 0$ then $[\, p_k \,]_i = [\, d_k \,]_i$. Otherwise, $[\, d_k \,]_i < 0$ with $\nabla_i f(x_k)[\, d_k \,]_i \ge 0$, and setting $[\, p_k \,]_i = 0$ makes the directional derivative more negative.

The working set at x_k is a subset of the *extended active set*, which is defined as

$$\mathcal{A}_{\epsilon_k}(x_k) = \{\, i : [\, x_k \,]_i \le \ell_i + \epsilon_k \ \text{ or } \ [\, x_k \,]_i \ge u_i - \epsilon_k \,\}.$$

It is shown in Section 3.3.2 that, under certain conditions, $\{\, \epsilon_k \,\} \to 0$, and $\mathcal{A}_{\epsilon_k}(x_k) = \mathcal{A}(x_k)$ for k sufficiently large, which would imply that $p_k = d_k$ for k sufficiently large.

A general projected-search method based on the proposed framework is summarized in Algorithm 3. There are various choices for the direction d_k. For example, if $d_k = -\mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)$, then the method is a variant of projected gradient descent. Other choices include computing d_k as the solution of the subproblem

$$\underset{d}{\text{minimize}}\ \nabla f(x_k)^{\mathrm{T}} d + \tfrac{1}{2} d^{\mathrm{T}} H_k d \quad \text{subject to } d_i = 0 \ \text{ for all } \ i \in \mathcal{W}_k, \tag{3.6}$$

where H_k a positive-definite approximation of $\nabla^2 f(x_k)$. The new method UBOPT presented in Section 3.4.2 computes d_k as the solution of (3.6) with H_k chosen as a positive-definite limited-memory BFGS approximation of $\nabla^2 f(x_k)$.

Algorithm 3 A class of active-set projected-search methods

```
constant: $\epsilon > 0$;
Choose $x_0 \in \mathit{\Omega}$;
Let $\epsilon_0 = \epsilon$;   $k = 0$;
while not converged do
    Determine the working set $\mathcal{W}_k$ (3.5);
    Compute a feasible descent direction $d_k$ at $x_k$ such that $[\, d_k \,]_i = 0$ if $i \in \mathcal{W}_k$;
    Modify $d_k$ to give a search direction $p_k$:
    $[\, p_k \,]_i = \begin{cases} \max\{[\, d_k \,]_i, 0\} & \text{if } [\, x_k \,]_i \le \ell_i + \epsilon_k, \\ \min\{[\, d_k \,]_i, 0\} & \text{if } [\, x_k \,]_i \ge u_i - \epsilon_k, \\ [\, d_k \,]_i & \text{otherwise;} \end{cases}$
    Compute a quasi-Wolfe step $\alpha_k$;   $x_{k+1} = \mathbf{proj}_{\mathit{\Omega}}(x_k + \alpha_k p_k)$;
    $\epsilon_{k+1} = \min\{\epsilon, \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\|\}$;
    $k \leftarrow k + 1$;
end while
```

3.3.2 Convergence analysis

In this section, the convergence properties of the class of projected-search active-set methods are considered. As an introduction, the convergence of a method with a quasi-Armijo search is examined first.

Theorem 3.3.1 (Active-set projected search with a quasi-Armijo search)**.** *Let f be a scalar-valued continuously differentiable function defined on $\mathit{\Omega} = \{\, x \in \mathbb{R}^n : \ell \le x \le u \,\}$. Assume that $x_0 \in \mathit{\Omega}$ is chosen such that the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded, and $\{\, x_k \,\}$ is defined by $x_{k+1} = x_k(\alpha_k)$, where α_k is a quasi-Armijo step. For an arbitrarily fixed $\epsilon > 0$, define $\epsilon_0 = \epsilon$, and*

$$\epsilon_k = \min\big\{\epsilon, \big\|\mathit{\Pi}_{k-1}^{\mathrm{T}} \nabla f(x_{k-1})\big\|\big\}$$

for $k \geq 1$, where each $\varPi_k$ is a matrix with orthonormal columns that spans the set of projected directions with respect to the working set $\mathcal{W}_k$. If $\{\, p_k \,\}$ is a sequence of descent directions with $\|p_k\| \leq \theta$ for some constant θ independent of k, $\varPi_k \varPi_k^{\mathrm{T}} p_k = p_k$ for all k, and the components of p_k satisfy $[\, p_k \,]_i \geq 0$ if $[\, x_k \,]_i \leq \ell_i + \epsilon_k$, and $[\, p_k \,]_i \leq 0$ if $[\, x_k \,]_i \geq u_i - \epsilon_k$, then

$$\lim_{k\to\infty} \left|\nabla f(x_k)^{\mathrm{T}} p_k\right| = 0.$$

Proof. First, we show that $\lim_{k\to\infty} \left|\nabla f(x_k)^{\mathrm{T}} p_k\right| = 0$ if $\liminf_{k\to\infty} \left\|\varPi_k^{\mathrm{T}} \nabla f(x_k)\right\| \neq 0$. Observe that the quasi-Armijo condition (3.1) implies that $\{\, f(x_k) \,\}$ is a strictly decreasing sequence. As the set $\mathcal{L}\big(f(x_0)\big)$ is bounded, it follows that $\{\, f(x_k) \,\}$ converges, with

$$0 = \lim_{k\to\infty} f(x_k) - f(x_{k+1}) \geq \lim_{k\to\infty} \alpha_k \eta_A |\nabla f(x_k)^{\mathrm{T}} p_k| = 0.$$

The proof is by contradiction. Suppose that $|\nabla f(x_k)^{\mathrm{T}} p_k| \not\to 0$ as $k \to \infty$, then there must exist some $\bar{\epsilon} > 0$ such that $|\nabla f(x_k)^{\mathrm{T}} p_k| > \bar{\epsilon}$ infinitely often. Let $\mathcal{G} = \{\, k : |\nabla f(x_k)^{\mathrm{T}} p_k| > \bar{\epsilon} \,\}$, then it must be that $\alpha_k \to 0$ for $k \in \mathcal{G}$. For all $k \in \mathcal{G}$, define the step $\beta_k = \alpha_k/\sigma$. The hypothesis that $\liminf_{k\to\infty} \left\|\varPi_k^{\mathrm{T}} \nabla f(x_k)\right\| \neq 0$ implies $\liminf_{k\to\infty} \epsilon_k > 0$. As $\{\, \|p_k\| \,\}$ is uniformly bounded by θ and $\liminf_{k\to\infty} \epsilon_k > 0$, there exists $\bar{k}$ such that each component of $\beta_k p_k$ satisfies $|[\, \beta_k p_k \,]_i| < \epsilon_k$ for all $k \geq \bar{k}$ in $\mathcal{G}$. The assumptions on components of p_k imply that $[\, p_k \,]_i > 0$ only if $u_i - [\, x_k \,]_i > \epsilon_k$, and $[\, p_k \,]_i < 0$ only if $[\, x_k \,]_i - \ell_i > \epsilon_k$. It follows that for all $k \geq \bar{k}$ in $\mathcal{G}$, $\ell_i \leq [\, x_k + \beta_k p_k \,]_i \leq u_i$ and $\mathbf{proj}_{\varOmega}(x_k + \beta_k p_k) = x_k + \beta_k p_k$.

Let $\bar{\mathcal{G}}$ denote the indices $k \geq \bar{k}$ of iterations at which a reduction in the initial step length was necessary, i.e., $\bar{\mathcal{G}} = \{\, k : t_k > 0,\ k \in \mathcal{G},\ k \geq \bar{k} \,\}$. Since α_k converges to zero, $\bar{\mathcal{G}}$ must be an infinite set. By definition,

$$f(x_k + \beta_k p_k) = f\big(\mathbf{proj}_{\varOmega}(x_k + \beta_k p_k)\big) > f(x_k) + \beta_k \eta_A \nabla f(x_k)^{\mathrm{T}} p_k, \ \text{ for all } \ k \in \bar{\mathcal{G}}.$$

Adding $-\beta_k \nabla f(x_k)^{\mathrm{T}} p_k$ to both sides and rearranging gives

$$\begin{aligned} f(x_k + \beta_k p_k) - f(x_k) - \beta_k \nabla f(x_k)^{\mathrm{T}} p_k &> -\beta_k (1 - \eta_A) \nabla f(x_k)^{\mathrm{T}} p_k \\ &> \beta_k (1 - \eta_A) \bar{\epsilon}, \ \text{ for all } \ k \in \bar{\mathcal{G}}. \end{aligned} \tag{3.7}$$

The Taylor expansion of $f(x_k + \beta_k p_k)$ gives

$$f(x_k + \beta_k p_k) - f(x_k) - \beta_k \nabla f(x_k)^{\mathrm{T}} p_k = \beta_k \int_0^1 \big(\nabla f(x_k + \tau \beta_k p_k) - \nabla f(x_k)\big)^{\mathrm{T}} p_k \, d\tau. \tag{3.8}$$

If $\|\cdot\|_D$ denotes the norm dual to $\|\cdot\|$, i.e., $\|x\|_D = \max_{v\neq 0} |x^{\mathrm{T}} v|/\|v\|$, then

$$\big|\big(\nabla f(x_k+\tau\beta_k p_k)-\nabla f(x_k)\big)^{\mathrm{T}} p_k\big| \le \|\nabla f(x_k+\tau\beta_k p_k)-\nabla f(x_k)\|_D\|p_k\|.$$

If this inequality is substituted in (3.8), it then follows from (3.7) that

$$\begin{aligned}(1-\eta_A)\bar{\epsilon} &< \int_0^1 \big(\nabla f(x_k+\tau\beta_k p_k)-\nabla f(x_k)\big)^{\mathrm{T}} p_k\, d\tau\\ &\le \max_{0\le\tau\le 1}\|\nabla f(x_k+\tau\beta_k p_k)-\nabla f(x_k)\|_D\|p_k\|, \quad \text{for all } \ k\in\bar{\mathcal{G}}.\end{aligned}$$

The continuity of ∇f implies that there exists some $\tau_k \in [0,\beta_k]$ such that

$$\max_{0\le\tau\le 1}\|\nabla f(x_k+\tau\beta_k p_k)-\nabla f(x_k)\|_D = \|\nabla f(x_k+\tau_k p_k)-\nabla f(x_k)\|_D.$$

Then

$$(1-\eta_A)\bar{\epsilon} < \|\nabla f(x_k+\tau_k p_k)-\nabla f(x_k)\|_D\|p_k\|. \tag{3.9}$$

However, $\alpha_k p_k \to 0$ implies $\tau_k p_k \to 0$ for $k\in\mathcal{G}$, and the continuity of ∇f gives

$$\|\nabla f(x_k+\tau_k p_k)-\nabla f(x_k)\|_D \to 0.$$

As $\{\,\|p_k\|\,\}$ is uniformly bounded above by θ, the right-hand side of (3.9) converges to zero, which gives the required contradiction.

Next it will be shown by contradiction that each convergent subsequence of $\{\,|\nabla f(x_k)^{\mathrm{T}} p_k|\,\}$ converges to zero regardless of the value of $\liminf_{k\to\infty}\big\|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\big\|$. As $\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} p_k = p_k$ for all k,

$$|\nabla f(x_k)^{\mathrm{T}} p_k| = |\nabla f(x_k)^{\mathrm{T}}\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} p_k| \tag{3.10}$$

for all k. Suppose that there exists a convergent subsequence of $\{\,|\nabla f(x_k)^{\mathrm{T}} p_k|\,\}$, say $\{\,|\nabla f(x_{k_j})^{\mathrm{T}} p_{k_j}|\,\}$, that converges to a positive value. Then, (3.10) implies that the sequence $\{\,|\nabla f(x_{k_j})^{\mathrm{T}}\mathit{\Pi}_{k_j}\mathit{\Pi}_{k_j}^{\mathrm{T}} p_{k_j}|\,\}$ converges to a positive value. As $\{\,\|p_k\|\,\}$ is bounded by a constant θ,

$$\liminf_{j\to\infty}\big\|\mathit{\Pi}_{k_j}^{\mathrm{T}}\nabla f(x_{k_j})\big\| > 0.$$

Applying the previous arguments to the subsequence $\{\, |\nabla f(x_{k_j})^{\mathrm{T}} p_{k_j}| \,\}$ gives

$$\lim_{j\to\infty} |\nabla f(x_{k_j})^{\mathrm{T}} p_{k_j}| = 0,$$

which is a contradiction.

As the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded, $\{\, |\nabla f(x_k)^{\mathrm{T}} p_k| \,\}$ must be a bounded sequence. It follows that

$$\liminf_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = \limsup_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = 0.$$

Therefore, $\lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = 0$. □

Theorem 3.3.2 (Active-set projected search with a quasi-Wolfe search)**.** *Let f be a scalar-valued continuously differentiable function defined on $\varOmega = \{\, x \in \mathbb{R}^n : \ell \le x \le u \,\}$. Assume that $x_0 \in \varOmega$ is chosen such that the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded, and $\{\, x_k \,\}$ is given by $x_{k+1} = x_k(\alpha_k)$, where α_k is a quasi-Wolfe step. For an arbitrarily fixed $\epsilon > 0$, define $\epsilon_0 = \epsilon$, and*

$$\epsilon_k = \min\big\{\epsilon, \big\|\varPi_{k-1}^{\mathrm{T}} \nabla f(x_{k-1})\big\|\big\}.$$

for $k \ge 1$, where each $\varPi_k$ is a matrix with orthonormal columns that spans the set of projected directions with respect to the working set $\mathcal{W}_k$. If $\{\, p_k \,\}$ is a sequence of descent directions with $\|p_k\| \le \theta$ for some constant θ independent of k, $\varPi_k \varPi_k^{\mathrm{T}} p_k = p_k$ for all k, and the components of p_k satisfy $[\, p_k \,]_i \ge 0$ if $[\, x_k \,]_i \le \ell_i + \epsilon_k$, and $[\, p_k \,]_i \le 0$ if $[\, x_k \,]_i \ge u_i - \epsilon_k$, then

$$\lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = 0.$$

Proof. First, we show that $\lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = 0$ if $\liminf_{k\to\infty} \big\|\varPi_k^{\mathrm{T}} \nabla f(x_k)\big\| \ne 0$. The first quasi-Wolfe condition ($\mathbf{C}_1$) is equivalent to the quasi-Armijo condition, and the arguments in the proof of Theorem 3.3.1 may be used to show that $\{\, f(x_k) \,\}$ is a convergent sequence. This implies that

$$\lim_{k\to\infty} \alpha_k \nabla f(x_k)^{\mathrm{T}} p_k = 0.$$

The proof is by contradiction. Suppose that $|\nabla f(x_k)^{\mathrm{T}} p_k| \not\to 0$ as $k \to \infty$, then there exists some $\bar\epsilon > 0$ such that $|\nabla f(x_k)^{\mathrm{T}} p_k| > \bar\epsilon$ infinitely often. Let $\mathcal{G} = \{\, k : |\nabla f(x_k)^{\mathrm{T}} p_k| > \bar\epsilon \,\}$, then it must be that $\alpha_k \to 0$ for $k \in \mathcal{G}$. As $\{\, \|p_k\| \,\}$ is uniformly bounded above by θ, $\alpha_k p_k \to 0$ for $k \in \mathcal{G}$.

If the quasi-Wolfe condition ($\mathbf{C}_2$) is satisfied, then

$$\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) \geq -\eta_{W} |\nabla f(x_k)^{\mathrm{T}} p_k|.$$

Similarly, if the quasi-Wolfe condition ($\mathbf{C}_4$) is satisfied, then

$$\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) \geq 0 \geq -\eta_{W} |\nabla f(x_k)^{\mathrm{T}} p_k|.$$

In either case, as $\nabla f(x_k)^{\mathrm{T}} p_k < 0$, it must hold that

$$\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k \geq (1-\eta_{W}) |\nabla f(x_k)^{\mathrm{T}} p_k| > (1-\eta_{W})\bar{\epsilon}, \ \text{ for } k \in \mathcal{G}.$$

The application of the triangle inequality yields

$$\begin{aligned} 0 < (1-\eta_{W})\bar{\epsilon} &< \left|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\right| \\ &\leq \left|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k)\right| \\ &\qquad + \left|\nabla f(x_k)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\right|. \end{aligned} \tag{3.11}$$

Let $\|\cdot\|_{D}$ denote the norm dual to $\|\cdot\|$, then

$$\begin{aligned} &\left|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k)\right| \\ &\qquad \leq \|\nabla f\big(x_k(\alpha_k)\big) - \nabla f(x_k)\|_{D} \|P_{x_k(\alpha_k)}(p_k)\| \leq \|\nabla f\big(x_k(\alpha_k)\big) - \nabla f(x_k)\|_{D} \|p_k\|. \end{aligned}$$

As ∇f is continuous and $\|p_k\|$ is uniformly bounded, the right-hand side of this inequality must converge to zero for $k \in \mathcal{G}$, which implies that

$$\left|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k)\right| \to 0, \ \text{ for } k \in \mathcal{G}.$$

Basic norm inequalities give

$$\begin{aligned} \left|\nabla f(x_k)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\right| &\leq \|\nabla f(x_k)\|_{D} \|P_{x_k(\alpha_k)}(p_k) - p_k\| \\ &= \|\nabla f(x_k)\|_{D} \|P_{x_k(\alpha_k)}(p_k) - P_{x_k}(p_k)\|. \end{aligned}$$

As the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded, and the gradient ∇f is continuous, the sequence of dual norms $\big\{\, \|\nabla f(x_k)\|_{D} \,\big\}$ is uniformly bounded. The hypothesis that $\liminf_{k\to\infty} \big\|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\big\| \neq 0$

implies $\liminf_{k\to\infty} \epsilon_k > 0$. Also, because

$$\|x_k(\alpha_k) - x_k\| \le \|\alpha_k p_k\| \to 0, \quad \text{for } k \in \mathcal{G},$$

there must exist an $\bar{k}$ such that for all $k \ge \bar{k}$ in $\mathcal{G}$,

$$[\, x_k(\alpha_k) - x_k \,]_i < \epsilon_k.$$

From the assumptions on the components of p_k, it must hold that for all $k \ge \bar{k}$ in $\mathcal{G}$, $[\, p_k \,]_i < 0$ only if $[\, x_k \,]_i > \ell_i + \epsilon_k$, in which case $[\, x_k(\alpha_k) \,]_i > \ell_i$; and $[\, p_k \,]_i > 0$ only if $[\, x_k \,]_i < u_i - \epsilon_k$, in which case $[\, x_k(\alpha_k) \,]_i < u_i$. It follows that, for $k \in \mathcal{G}$ sufficiently large,

$$P_{x_k(\alpha_k)}(p_k) = P_{x_k}(p_k) = p_k.$$

Therefore,

$$\|\nabla f(x_k)\|_D \|P_{x_k(\alpha_k)}(p_k) - P_{x_k}(p_k)\| \to 0, \quad \text{for } k \in \mathcal{G},$$

and consequently

$$\left|\nabla f(x_k)^{\mathrm{T}} P_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\right| \to 0, \quad \text{for } k \in \mathcal{G}.$$

It follows that the right-hand side of (3.11) converges to zero for $k \in \mathcal{G}$, which gives the required contradiction.

It remains to consider the case where the quasi-Wolfe condition ($\mathbf{C}_3$) is satisfied, i.e.,

$$\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{x_k(\alpha_k)}(p_k) \ge -\eta_W |\nabla f(x_k)^{\mathrm{T}} p_k|.$$

The assumption that $\nabla f(x_k)^{\mathrm{T}} p_k < 0$ gives

$$\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k \ge (1-\eta_W)|\nabla f(x_k)^{\mathrm{T}} p_k| > (1-\eta_W)\bar{\epsilon}, \quad \text{for } k \in \mathcal{G},$$

which implies that

$$\begin{aligned} 0 < (1-\eta_W)\bar{\epsilon} &< \left|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\right| \\ &\le \left|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} P^{-}_{x_k(\alpha_k)}(p_k)\right| \\ &\qquad + \left|\nabla f(x_k)^{\mathrm{T}} P^{-}_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\right|. \end{aligned} \tag{3.12}$$

The definition of the dual norm yields

$$\begin{aligned}\Big|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P^-_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} P^-_{x_k(\alpha_k)}(p_k)\Big| \\ \le \|\nabla f\big(x_k(\alpha_k)\big) - \nabla f(x_k)\|_D \|P^-_{x_k(\alpha_k)}(p_k)\| \le \|\nabla f\big(x_k(\alpha_k)\big) - \nabla f(x_k)\|_D \|p_k\|.\end{aligned}$$

From the continuity of ∇f and uniform boundedness of $\|p_k\|$, the right-hand side of the above inequality converges to zero for $k \in \mathcal{G}$, which means that

$$\Big|\nabla f\big(x_k(\alpha_k)\big)^{\mathrm{T}} P^-_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} P^-_{x_k(\alpha_k)}(p_k)\Big| \to 0, \text{ for } k \in \mathcal{G}.$$

Also,

$$\begin{aligned}\Big|\nabla f(x_k)^{\mathrm{T}} P^-_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\Big| &\le \|\nabla f(x_k)\|_D \|P^-_{x_k(\alpha_k)}(p_k) - p_k\| \\ &= \|\nabla f(x_k)\|_D \|P^-_{x_k(\alpha_k)}(p_k) - P_{x_k}(p_k)\|.\end{aligned}$$

As the level set $\mathcal{L}\big(f(x_0)\big)$ is bounded, and ∇f is continuous, it must hold that the sequence of dual norms $\{\, \|\nabla f(x_k)\|_D \,\}$ is uniformly bounded. Also, as

$$\|x_k(\alpha_k) - x_k\| \le \|\alpha_k p_k\| \to 0, \text{ for } k \in \mathcal{G},$$

arguments analogous to those used to establish convergence in cases $(\mathbf{C}_2)$ and $(\mathbf{C}_4)$ give

$$P^-_{x_k(\alpha_k)}(p_k) = P_{x_k}(p_k) = p_k \text{ for } k \in \mathcal{G} \text{ sufficiently large,}$$

in which case

$$\|\nabla f(x_k)\|_D \|P^-_{x_k(\alpha_k)}(p_k) - P_{x_k}(p_k)\| \to 0, \text{ for } k \in \mathcal{G}.$$

This implies that

$$\Big|\nabla f(x_k)^{\mathrm{T}} P^-_{x_k(\alpha_k)}(p_k) - \nabla f(x_k)^{\mathrm{T}} p_k\Big| \to 0, \text{ for } k \in \mathcal{G}.$$

It follows that the right-hand side of (3.12) converges to zero for $k \in \mathcal{G}$, which gives the required contradiction.

Finally, the same arguments from the proof of Theorem 3.3.1 imply that

$$\lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = 0$$

regardless of the value of $\liminf_{k\to\infty} \big\|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\big\|$. □

Based on the framework described in Section 3.3.1, the limit $\lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| = 0$ implies that

$$\lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} d_k| = 0, \tag{3.13}$$

which would further imply that the projected gradient, $\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)$, converges to zero for an appropriate choice of d_k. For example, if $d_k = -\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)$, or d_k is the solution of the subproblem (3.6) with the two-norm of the projected approximate Hessian, $\|\mathit{\Pi}_k^{\mathrm{T}} H_k \mathit{\Pi}_k\|$, uniformly bounded, then it may be verified that (3.13) implies that $\|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\| \to 0$.

Under the nondegeneracy assumption defined below, any algorithm based on the proposed framework for which $\|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\| \to 0$ will identify the optimal active set in a finite number of iterations.

Definition 3.3.1. *A point $x^* \in \Omega$ is a* stationary point *of* (BC) *if $\nabla_i f(x^*) = 0$ for $\ell_i < x_i^* < u_i$, $\nabla_i f(x^*) \ge 0$ for $x_i^* = \ell_i$ and $\ell_i < u_i$, and $\nabla_i f(x^*) \le 0$ for $x_i^* = u_i$ and $\ell_i < u_i$. A stationary point x^* is* nondegenerate *if $\nabla_i f(x^*) > 0$ for $x_i^* = \ell_i$ and $\ell_i < u_i$, and $\nabla_i f(x^*) < 0$ for $x_i^* = u_i$ and $\ell_i < u_i$.*

The next result shows that a projected-search method with either a quasi-Armijo or quasi-Wolfe search will identify the optimal active set in a finite number of iterations.

Theorem 3.3.3. *In addition to the assumptions of* Theorem 3.3.1 *or* Theorem 3.3.2*, assume that $\{\, x_k \,\}$ converges to a nondegenerate stationary point x^*. Consider the extended active set*

$$\mathcal{A}_{\epsilon_k}(x_k) = \big\{\, i : [\, x_k\,]_i \le \ell_i + \epsilon_k \;\; \textit{or} \;\; [\, x_k\,]_i \ge u_i - \epsilon_k \,\big\}.$$

If $\|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\| \to 0$, then $\mathcal{A}_{\epsilon_k}(x_k) = \mathcal{A}(x_k) = \mathcal{A}(x^*)$ for all k sufficiently large.

Proof. First, we show that $\mathcal{A}(x^*) \subset \mathcal{A}_{\epsilon_k}(x_k)$ for k sufficiently large by contradiction. Assume the opposite is true, then there exists $i \in \mathcal{A}(x^*)$ such that $i \notin \mathcal{A}_{\epsilon_k}(x_k)$ for an infinite subsequence $\mathcal{K}$, which implies that $i \notin \mathcal{W}_k$ for all $k \in \mathcal{K}$. It follows that

$$|\nabla_i f(x_k)| \le \|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\| \;\text{ for } k \in \mathcal{K}.$$

As f is continuously differentiable and $\|\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)\| \to 0$, letting $k \to \infty$ in $\mathcal{K}$ gives

$$|\nabla_i f(x^*)| = \lim_{k\to\infty, k\in\mathcal{K}} |\nabla_i f(x_k)| = 0.$$

This contradicts the nondegeneracy of x^*.

Now we show that $\mathcal{A}_{\epsilon_k}(x_k) \subset \mathcal{A}(x^*)$ for k sufficiently large. If $\ell_i = u_i$, a simple argument gives $i \in \mathcal{A}_{\epsilon_k}(x_k)$ and $i \in \mathcal{A}(x^*)$. Consider an index i such that $\ell_i < u_i$. From the definition of ϵ_k, the assumption $\|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\| \to 0$ implies that $\epsilon_k \to 0$. Hence, for k sufficiently large, $\ell_i + \epsilon_k < u_i - \epsilon_k$. If $i \notin \mathcal{A}(x^*)$, then $\ell_i < [\, x^* \,]_i < u_i$. As $\{\, x_k \,\} \to x^*$ and $\epsilon_k \to 0$, $\ell_i + \epsilon_k < [\, x_k \,]_i < u_i - \epsilon_k$ for k sufficiently large, which implies that $i \notin \mathcal{A}_{\epsilon_k}(x_k)$. Therefore, if $i \notin \mathcal{A}(x^*)$, then $i \notin \mathcal{A}_k(x_k)$, i.e. $\mathcal{A}_{\epsilon_k}(x_k) \subset \mathcal{A}(x^*)$ for k sufficiently large. We conclude that $\mathcal{A}_{\epsilon_k}(x_k) = \mathcal{A}(x^*)$ for all k sufficiently large.

It remains to show that $\mathcal{A}(x_k) = \mathcal{A}_{\epsilon_k}(x_k)$ for k sufficiently large. Obviously $\mathcal{A}(x_k) \subset \mathcal{A}_{\epsilon_k}(x_k)$ for all k. It is trivial if $\ell_i = u_i$. Now consider the case where $\ell_i < u_i$. Note that $\{\, x_k \,\} \to x^*$ implies $\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0$. As $\lim_{k\to\infty}(u_i - \epsilon_{k+1}) - (\ell_i + \epsilon_k) = u_i - \ell_i > 0$, $|[\, x_{k+1} - x_k \,]_i| < (u_i - \epsilon_{k+1}) - (\ell_i + \epsilon_k)$ for k sufficiently large. Suppose k_0 is such that, for all $k \ge k_0$, $\mathcal{A}_{\epsilon_k}(x_k) = \mathcal{A}(x^*)$ and $|[\, x_{k+1} - x_k \,]_i| < (u_i - \epsilon_{k+1}) - (\ell_i + \epsilon_k)$. The inclusion $\mathcal{A}_{\epsilon_k}(x_k) \subset \mathcal{A}(x_k)$ for all $k \ge k_0$ is established using a contradiction argument. Assume that there exists $i \in \mathcal{A}_{\epsilon_k}(x_k) = \mathcal{A}(x^*)$ for all $k \ge k_0$, but $i \notin \mathcal{A}(x_k)$ for some $\bar{k} \ge k_0$. Then either $\ell_i < [\, x_{\bar{k}} \,]_i \le \ell_i + \epsilon_{\bar{k}}$ or $u_i - \epsilon_{\bar{k}} \le [\, x_{\bar{k}} \,]_i < u_i$. If the inequality $\ell_i < [\, x_{\bar{k}} \,]_i \le \ell_i + \epsilon_{\bar{k}}$ holds, the definition of p_k in Algorithm 3 implies that $[\, p_{\bar{k}} \,]_i \ge 0$, and it must be the case that $\ell_i < [\, x_{\bar{k}} \,]_i \le [\, x_{\bar{k}+1} \,]_i$. In addition, $\big|[\, x_{\bar{k}+1} - x_{\bar{k}} \,]_i\big| < (u_i - \epsilon_{\bar{k}+1}) - (\ell_i + \epsilon_{\bar{k}})$ implies that $[\, x_{\bar{k}+1} \,]_i < u_i - \epsilon_{\bar{k}+1}$. As $i \in \mathcal{A}_{\epsilon_{\bar{k}+1}}(x_{\bar{k}+1})$, it must hold that $\ell_i < [\, x_{\bar{k}} \,]_i \le [\, x_{\bar{k}+1} \,]_i \le \ell_i + \epsilon_{\bar{k}+1}$. Inductively, for all $k \ge \bar{k}$, $\ell_i < [\, x_{\bar{k}} \,]_i \le [\, x_k \,]_i \le \ell_i + \epsilon_k$, which implies that $[\, x^* \,]_i \ge [\, x_{\bar{k}} \,]_i > \ell_i$. A similar argument shows that if $u_i - \epsilon_{\bar{k}} \le [\, x_{\bar{k}} \,]_i < u_i$, then $[\, x^* \,]_i \le [\, x_{\bar{k}} \,]_i < u_i$. It follows that $i \notin \mathcal{A}(x^*)$, which contradicts the assumption that $i \in \mathcal{A}_{\epsilon_k}(x_k) = \mathcal{A}(x^*)$ for all $k \ge k_0$. Therefore, $\mathcal{A}_{\epsilon_k}(x_k) \subset \mathcal{A}(x_k)$ for all $k \ge k_0$, which completes the proof. □

A simple example shows that the nondegeneracy of a stationary point is necessary for identifying the optimal active set in a finite number of iterations. Let $f : \mathbb{R}^2 \to \mathbb{R}$ be given by $f(x) = \frac{1}{5}\|x\|^2$, and let $\mathit{\Omega} = \{\, x \in \mathbb{R}^2 : x \ge 0 \,\}$. For this problem $x^* = (0,0)^{\mathrm{T}}$ is a degenerate stationary point and the global minimizer of f over $\mathit{\Omega}$. Assume that the step length $\alpha_k \le 1$ for all k, and let $\epsilon = \frac{1}{\sqrt{2}}$. Starting from $x_0 = (1,1)^{\mathrm{T}}$, the projected-gradient method gives

$$x_k = \prod_{j=0}^{k} (1 - \tfrac{2}{5}\alpha_j) \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \text{and} \quad \epsilon_k = \tfrac{2}{5}\|x_{k-1}\| = \tfrac{2\sqrt{2}}{5} \prod_{j=0}^{k-1} \big(1 - \tfrac{2}{5}\alpha_j\big)$$

for $k \ge 1$. Then $\{\, x_k \,\}$ converges to the degenerate stationary point x^*, and

$$[\, x_k \,]_i = \prod_{j=0}^{k} (1 - \tfrac{2}{5}\alpha_j) > \tfrac{2\sqrt{2}}{5} \prod_{j=0}^{k-1} \bigl(1 - \tfrac{2}{5}\alpha_j\bigr) = \epsilon_k, \quad i = 1, 2$$

for all $k \ge 1$. It follows that $\mathcal{A}_{\epsilon_k}(x_k) = \emptyset$ for all k, although $\mathcal{A}(x^*) = \{\, 1, 2 \,\}$.

3.4 A Limited-Memory Reduced-Hessian Method

Based on the general framework of projected-search active-set methods described previously, a quasi-Newton projected-search method UBOPT for unconstrained and bound-constrained optimization can now be proposed, which is an extension of the limited-memory reduced-Hessian method for unconstrained optimization of Leonard [66] and Gill and Leonard [47].

3.4.1 Background: an L-RHR method for unconstrained optimization

This section gives a brief review of the limited-memory reduced-Hessian method L-RHR for the unconstrained minimization of the twice continuously differentiable function $f : \mathbb{R}^n \mapsto \mathbb{R}$. For more details, see Gill and Leonard [47]. As a quasi-Newton method, it generates a sequence of iterates $\{\, x_k \,\}$ such that $x_{k+1} = x_k + \alpha_k p_k$, where p_k is a descent direction and α_k is a Wolfe step. The search direction satisfies $H_k p_k = -\nabla f(x_k)$, where H_k is a positive-definite approximation to the Hessian matrix of f. Given H_0, the sequence of approximate Hessian matrices $\{\, H_k \,\}$ is generated using the BFGS update:

$$H_{k+1} = H_k - \frac{1}{s_k^{\mathrm{T}} H_k s_k} H_k s_k s_k^{\mathrm{T}} H_k + \frac{1}{w_k^{\mathrm{T}} s_k} w_k w_k^{\mathrm{T}}, \tag{3.14}$$

where $s_k = x_{k+1} - x_k$, $w_k = \nabla f(x_{k+1}) - \nabla f(x_k)$, and $w_k^{\mathrm{T}} s_k$ approximates the curvature of f along p_k.

The *reduced-Hessian method* of Gill and Leonard takes advantage of the implicit structure of the quasi-Newton Hessian to compute search directions from a smaller search space. The method is implemented in a limited-memory framework by limiting the number of basis vectors for the search space. The *gradient subspace* defined as span$\{\, \nabla f(x_0),\ \nabla f(x_1),\ \ldots,\ \nabla f(x_k) \,\}$ and denoted by $\mathcal{G}_k$, with $\mathcal{G}_k^{\perp}$ denoting the orthogonal complement of $\mathcal{G}_k$ in $\mathbb{R}^n$. Reduced-Hessian methods are based on the following result (see, e.g., Fletcher and Powell [32], Fenelon [27], and Siegel [81]).

Lemma 3.4.1. *Consider the BFGS method applied to a general nonlinear function. If $H_0 = \sigma I$ $(\sigma > 0)$ and $H_k p_k = -\nabla f(x_k)$, then $p_k \in \mathcal{G}_k$ for all k. Moreover, if $z \in \mathcal{G}_k$ and $y \in \mathcal{G}_k^\perp$, then $H_k z \in \mathcal{G}_k$ and $H_k y = \sigma y$.*

If r_k denotes the dimension of $\mathcal{G}_k$, let Z_k be an $n \times r_k$ matrix whose columns form an orthonormal basis for $\mathcal{G}_k$. Given an $(n-r_k)\times n$ orthonormal basis Y_k for $\mathcal{G}_k^\perp$, the matrix $Q_k = \begin{pmatrix} Z_k & Y_k \end{pmatrix}$ defines an orthogonal transformation $x \mapsto Q_k x$. The transformed gradient and approximate Hessian are then given by $Q_k^{\mathrm{T}} \nabla f(x_k)$ and $Q_k^{\mathrm{T}} H_k Q_k$, respectively. If $H_0 = \sigma I$ $(\sigma > 0)$, it follows from Lemma 3.4.1 that the transformation induces a block-diagonal structure, with

$$Q_k^{\mathrm{T}} H_k Q_k = \begin{pmatrix} Z_k^{\mathrm{T}} H_k Z_k & 0 \\ 0 & \sigma I_{n-r_k} \end{pmatrix} \quad \text{and} \quad Q_k^{\mathrm{T}} \nabla f(x_k) = \begin{pmatrix} Z_k^{\mathrm{T}} \nabla f(x_k) \\ 0 \end{pmatrix}. \tag{3.15}$$

The matrix $Z_k^{\mathrm{T}} H_k Z_k$ is positive-definite and is known as the approximate reduced Hessian (or just reduced Hessian). The vector $Z_k^{\mathrm{T}} \nabla f(x_k)$ is known as the reduced gradient. If the equation $H_k p_k = -\nabla f(x_k)$ for the search direction is written as $(Q_k^{\mathrm{T}} H_k Q_k) Q_k^{\mathrm{T}} p_k = -Q_k^{\mathrm{T}} \nabla f(x_k)$, then it follows from (3.15) that

$$p_k = Z_k q_k, \quad \text{where } q_k \text{ satisfies} \quad Z_k^{\mathrm{T}} H_k Z_k q_k = -Z_k^{\mathrm{T}} \nabla f(x_k). \tag{3.16}$$

The matrices Z_k and $Z_k^{\mathrm{T}} H_k Z_k$ may be used to reconstruct H_k, which need not be stored explicitly. In particular, it satisfies that

$$\begin{aligned} H_k &= Q_k Q_k^{\mathrm{T}} H_k Q_k Q_k^{\mathrm{T}} \\ &= \begin{pmatrix} Z_k & Y_k \end{pmatrix} \begin{pmatrix} Z_k^{\mathrm{T}} H_k Z_k & 0 \\ 0 & \sigma I_{n-r_k} \end{pmatrix} \begin{pmatrix} Z_k^{\mathrm{T}} \\ Y_k^{\mathrm{T}} \end{pmatrix} \\ &= Z_k (Z_k^{\mathrm{T}} H_k Z_k) Z_k^{\mathrm{T}} + \sigma (I - Z_k Z_k^{\mathrm{T}}). \end{aligned} \tag{3.17}$$

If B_k is an $n \times r_k$ matrix with columns that form a basis for $\mathcal{G}_k$, an orthonormal basis Z_k can be defined in terms of the economy-size QR decomposition $B_k = Z_k T_k$, where T_k is a nonsingular $r_k \times r_k$ upper-triangular matrix. In practice, Z_k can be stored explicitly along with T_k, or implicitly by storing only B_k and T_k, with computations involving Z_k utilizing $Z_k = B_k T_k^{-1}$. If the Cholesky factorization $Z_k^{\mathrm{T}} H_k Z_k = R_k^{\mathrm{T}} R_k$ is known, q_k can be computed from the forward substitution $R_k^{\mathrm{T}} d_k = -Z_k^{\mathrm{T}} \nabla f(x_k)$ and back-substitution $R_k q_k = d_k$.

The dimension of $Z_k^{\mathrm{T}} H_k Z_k$ is limited by discarding the oldest basis vector when the number of basis vectors exceeds some predefined limit m. Assume for the moment that the gradients in the sequence span $\{\, \nabla f(x_0),\ \nabla f(x_1),\ \ldots,\ \nabla f(x_k)\, \}$ are linearly independent. Lemma 3.4.1 implies that the search direction p_k lies in $\mathcal{G}_k$ for all k. Siegel [81] proposed that a subset of $\{\, p_k\, \}$ be used to form a basis for $\mathcal{G}_k$ instead of $\{\, \nabla f(x_k)\, \}$ and showed that this modification endows the method with finite termination on a strictly convex quadratic function. Consider any iteration k such that $1 \le k \le m-1$. At the start of the iteration, the directions $p_0, \ldots, p_{k-1}$ are known, but p_k has yet to be computed from equations (3.16) that use Z_k. This implies that it is not possible to use p_k as part of B_k. Nevertheless, $\mathcal{G}_k$ is spanned by both the gradients and the search directions, which means that the latest gradient $\nabla f(x_k)$ can be used as a temporary basis vector until p_k has been computed, at which point it can be swapped with $\nabla f(x_k)$. The swap does not change Z_k, but the last column of T_k is replaced by the vector $q_k = Z_k^{\mathrm{T}} p_k$ found as part of the computation of p_k in (3.16). If $\nabla f(x_{k+1})$ is accepted after the line search, it is added to the basis and the QR factors are updated as in (3.18). This update expands the reduced Hessian by a row and column (see (3.19)), and the last diagonal is reinitialized with $\sigma_k = w_k^{\mathrm{T}} w_k / w_k^{\mathrm{T}} s_k$.

If $k \ge m-1$, the addition of $\nabla f(x_{k+1})$ gives a basis with $m+1$ columns and the oldest column p_{k-m+1} must be removed before starting iteration $k+1$. The factors Z_{k+1} and T_{k+1} associated with the next basis $B_{k+1} = \begin{pmatrix} p_{k-m+2} & \cdots & p_k & \nabla f(x_{k+1}) \end{pmatrix}$ are updated using two sets of plane rotations applied on the right of the orthogonal factor and left of the triangular factor of $\begin{pmatrix} p_{k-m+1} & \cdots & p_k & \nabla f(x_{k+1}) \end{pmatrix}$. Further details of the methods for updating the QR and Cholesky factors when a column is removed from the basis are given by Gill and Leonard [47].

During the k-th iteration of L-RHR, the number of columns in B_k (and Z_k) can either remain unchanged or increase by one, depending on whether or not the new gradient $\nabla f(x_{k+1})$ lies in $\mathcal{G}_k$. This is determined from the value of the scalar ρ_{k+1} such that $\rho_{k+1} = \|(I - Z_k Z_k^{\mathrm{T}})\nabla f(x_{k+1})\|$. If $\rho_{k+1} = 0$, then $\nabla f(x_{k+1}) \in \mathcal{G}_k$ and $\nabla f(x_{k+1})$ is said to be *rejected*. The matrix factors for the next iteration remain unchanged. Otherwise, $r_{k+1} = r_k + 1$ and $\nabla f(x_{k+1})$ is said to be *accepted*. In this case, B_k is augmented by a new column $\nabla f(x_{k+1})$, and the matrix factors of B_{k+1} are given by

$$B_{k+1} = \begin{pmatrix} B_k & \nabla f(x_{k+1}) \end{pmatrix} = \begin{pmatrix} Z_k & z_{k+1} \end{pmatrix} \begin{pmatrix} T_k & Z_k^{\mathrm{T}} \nabla f(x_{k+1}) \\ 0 & \rho_{k+1} \end{pmatrix} = Z_{k+1} T_{k+1}, \tag{3.18}$$

where z_{k+1} is defined by the identity $\rho_{k+1} z_{k+1} = (I - Z_k Z_k^{\mathrm{T}})\nabla f(x_{k+1})$. Note that T_{k+1} is nonsingular as $\rho_{k+1} \neq 0$. The Cholesky factor R_k is updated by adding a row and column to account for the new last column of Z_{k+1}. It follows from Lemmas 3.4.1 and (3.15) that

$$Z_{k+1}^{\mathrm{T}} H_k Z_{k+1} = \begin{pmatrix} Z_k^{\mathrm{T}} H_k Z_k & Z_k^{\mathrm{T}} H_k z_{k+1} \\ z_{k+1}^{\mathrm{T}} H_k Z_k & z_{k+1}^{\mathrm{T}} H_k z_{k+1} \end{pmatrix} = \begin{pmatrix} Z_k^{\mathrm{T}} H_k Z_k & 0 \\ 0 & \sigma \end{pmatrix}, \tag{3.19}$$

giving the expanded block-diagonal factor

$$R_k^{(1)} = \begin{pmatrix} R_k & 0 \\ 0 & \sigma^{1/2} \end{pmatrix}.$$

If $\nabla f(x_{k+1})$ is rejected, then $r_{k+1} = r_k$ and $R_k^{(1)} = R_k$.

In addition, the factor R_{k+1} is computed by modifying $R_k^{(1)}$ to reflect the rank-two BFGS update to $Z_{k+1}^{\mathrm{T}} H_k Z_{k+1}$ resulting from the rank-two update to H_k defined in (3.14). Let $s = Z_{k+1}^{\mathrm{T}} s_k$ and $y = Z_{k+1}^{\mathrm{T}} w_k$. If $u = R_k^{(1)} s / \|R_k^{(1)} s\|$ and $v = y/\sqrt{y^{\mathrm{T}} s} - R_k^{(1)\mathrm{T}} u$, then it may be verified by direct multiplication that

$$Z_{k+1}^{\mathrm{T}} H_{k+1} Z_{k+1} = (R_k^{(1)} + uv^{\mathrm{T}})^{\mathrm{T}} (R_k^{(1)} + uv^{\mathrm{T}}).$$

Two sets of plane rotations can be applied to restore $R_k^{(1)} + uv^{\mathrm{T}}$ to upper-triangular form. The first, S_1, is the product of plane rotations $P_{1,2} P_{1,3} \cdots P_{1,r_k}$ that zero out components 2 through r_k of u, i.e., $S_1 u = \gamma e_1$, with $\gamma = \pm\|u\|$. The application of S_1 to $R_k^{(1)} + uv^{\mathrm{T}}$ gives

$$S_1(R_k^{(1)} + uv^{\mathrm{T}}) = S_1 R_k^{(1)} + \gamma e_1 v^{\mathrm{T}}. \tag{3.20}$$

By construction, S_1 applied to $R_k^{(1)}$ results in an upper-Hessenberg matrix. As $\gamma e_1 v^{\mathrm{T}}$ is a matrix with only nonzeros in its first row, the right-hand side of (3.20) is also upper-Hessenberg. A second set of plane rotations S_2 is then defined such that $R_k^{(2)} = S_2 S_1 R_2$, where $S_2 = P'_{1,2} P'_{2,3} \cdots P'_{r_k-1,r_k}$. The resulting matrix $R_k^{(2)}$ is the upper-triangular factor of $Z_{k+1}^{\mathrm{T}} H_{k+1} Z_{k+1}$. For more details, see Dennis and Schnabel [20], and Gill and Leonard [47]).

In finite-precision arithmetic, the use of the economy QR factorization instead of the full QR may cause a loss of orthogonality in Z_k as columns are added to the basis. When a gradient is accepted, the new column is computed as $z_{k+1} = v_{k+1}/\rho_{k+1}$, where $v_{k+1} = (I - Z_k Z_k^{\mathrm{T}})\nabla f(x_{k+1})$ and $\rho_{k+1} = \|v_{k+1}\|$. This choice of z_{k+1} is designed to force $Z_k^{\mathrm{T}} v_{k+1}$ to be small relative to $\|\nabla f(x_{k+1})\|$. However, if ρ_{k+1} is small and $\|Z_k^{\mathrm{T}} v_{k+1}\| = \epsilon \|\nabla f(x_{k+1})\|$ for some small ϵ, then the normalized vector $z_{k+1} = v_{k+1}/\rho_{k+1}$ would satisfy only $\|Z_k^{\mathrm{T}} z_{k+1}\| = \epsilon \|\nabla f(x_{k+1})\|/\rho_{k+1}$. In this situation, the error relative to $\|\nabla f(x_{k+1})\|$ may be very large, resulting in a significant loss of orthogonality in the computed z_{k+1}. To rectify this loss of orthogonality, Daniel, Gragg, Kaufman and Stewart [19] propose a *reorthogonalization* scheme. If $\|v_{k+1}\|/\|\nabla f(x_{k+1})\|$ is small, then v_{k+1} is refined using the scheme

$$v'_{k+1} = (I - Z_k Z_k^{\mathrm{T}})v_{k+1}.$$

If $\|v'_{k+1}\|/\|v_{k+1}\|$ is not too small, then v'_{k+1} can be scaled to provide a satisfactory update to Z_{k+1}. Otherwise, the process is repeated.

The initial approximate Hessian can greatly influence the practical performance of quasi-Newton methods. A choice of $H_0 = \sigma I$, with some arbitrary positive σ can result in poor performance, especially when $\nabla^2 f(x^*)$ is ill-conditioned. Moreover, equation (3.15) reveals that σ represents the approximate curvature along all directions in $\mathcal{G}_k^{\perp}$. To enhance the performance of L-RHR, *Hessian reinitialization* is applied to "reset" the approximate Hessian matrix with current curvature information. When a new gradient is accepted, the reduced Hessian is expanded with σ_k rather than σ in equation (3.19). Gill and Leonard [47] use $\sigma_k = w_k^{\mathrm{T}} w_k / w_k^{\mathrm{T}} s_k$ in L-RHR.

3.4.2 UBOPT: an L-RHR method for bound constraints

The new algorithm UBOPT is introduced in this section as an extension of the algorithm L-RHR for solving problem (BC). Given an initial $x_0 \in \mathit{\Omega}$, the sequence of iterates $\{\, x_k \,\}$ is generated as described in Algorithm 3, with the search direction p_k computed in terms of a descent direction d_k that is determined by solving the following subproblem:

$$\underset{d}{\text{minimize}}\ \nabla f(x_k)^{\mathrm{T}} d + \tfrac{1}{2} d^{\mathrm{T}} H_k d \quad \text{subject to}\ \ d_i = 0\ \ \text{for all}\ \ i \in \mathcal{W}_k, \tag{3.21}$$

where H_k is a positive-definite limited-memory BFGS approximation of $\nabla^2 f(x_k)$, and $\mathcal{W}_k$ is the working set defined as in (3.5). The matrix H_k is maintained in reduced-Hessian form and is not stored explicitly.

The complement of $\mathcal{W}_k$ in $\{\, 1, 2, \ldots, n \,\}$ is denoted by $\mathcal{F}_k$, which may be regarded as the set of indices of the variables that are free to move at x_k. Let $\mathit{\Pi}_k$ denote a matrix with orthonormal columns that span the set of projected directions with respect to the working set $\mathcal{W}_k$. The columns of $\mathit{\Pi}_k$ can be taken as the columns of the identity matrix of order n associated with the indices in $\mathcal{F}_k$. If d_k is the solution of (3.21), then it satisfies that

$$\mathit{\Pi}_k^{\mathrm{T}} H_k d_k = -\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k).$$

A result analogous to Lemma 3.4.1 can be proved for the bound-constrained case.

Lemma 3.4.2. *Let d_k be the unique solution of* (3.21)*, and let*

$$\mathcal{S}_k = \operatorname{span}\left\{\{\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_j) : j = 0, \ldots, k\} \cup \{\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_j s_j : j = 0, \ldots, k-1\}\right\},$$

where $s_j = x_{j+1} - x_j$ for each j. If $H_0 = \sigma I$, then $d_k \in \mathcal{S}_k$. Moreover, if $z \in \mathcal{S}_k$, then $\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_k z \in \mathcal{S}_k$; if $y \in \mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\mathcal{S}_k^{\perp}$, then $H_k y = \sigma y$.

Proof. Recall the BFGS update formula

$$H_{k+1} = H_k - \frac{1}{s_k^{\mathrm{T}} H_k s_k} H_k s_k s_k^{\mathrm{T}} H_k + \frac{1}{w_k^{\mathrm{T}} s_k} w_k w_k^{\mathrm{T}},$$

where $s_k = x_{k+1} - x_k$, and $w_k = \nabla f(x_{k+1}) - \nabla f(x_k)$. It follows that

$$H_k = H_0 - \sum_{j=0}^{k-1} \left(\frac{1}{s_j^{\mathrm{T}} H_j s_j} H_j s_j s_j^{\mathrm{T}} H_j - \frac{1}{w_j^{\mathrm{T}} s_j} w_j w_j^{\mathrm{T}} \right). \tag{3.22}$$

If $H_0 = \sigma I$, then

$$\begin{aligned} d_k =& \mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} d_k \\ =& \frac{1}{\sigma}\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_0 d_k \\ =& \frac{1}{\sigma}\left[\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_k d_k + \sum_{j=0}^{k-1}\left(\frac{s_j^{\mathrm{T}} H_j d_k}{s_j^{\mathrm{T}} H_j s_j}\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_j s_j - \frac{w_j^{\mathrm{T}} d_k}{w_j^{\mathrm{T}} s_j}\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} w_j\right)\right] \\ =& \frac{1}{\sigma}\left[-\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k) + \sum_{j=0}^{k-1}\left(\frac{s_j^{\mathrm{T}} H_j d_k}{s_j^{\mathrm{T}} H_j s_j}\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_j s_j - \frac{w_j^{\mathrm{T}} d_k}{w_j^{\mathrm{T}} s_j}\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} w_j\right)\right]. \end{aligned}$$

Therefore, $d_k \in \mathcal{S}_k$. It follows from (3.22) that, if $z \in \mathcal{S}_k$, then $\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_k z \in \mathcal{S}_k$; and if $y \in \mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\mathcal{S}_k^{\perp}$, then $H_k y = H_k\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} y = \sigma y$. □

It is worth noting that, in the case when the search direction is not bent during the line search, i.e. $s_k = \alpha_k p_k$, then the vector $\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}} H_k s_k$ is parallel to the projected gradient $\mathit{\Pi}_k\mathit{\Pi}_k^{\mathrm{T}}\nabla f(x_k)$.

As described in the previous section, the L-RHR method for unconstrained optimization computes a search direction p_k that lies within a subspace

$$\bar{\mathcal{G}}_k \triangleq \operatorname{span}\left\{p_l, \cdots, p_{k-1}, \nabla f(x_k)\right\}, \quad \text{with } \; l = \max\{0, k-m+1\}.$$

For bound-constrained optimization, the computation of the search directions is based on the projected gradients. Additionally, to improve practical efficiency, the strategy proposed by Siegel [81] that swaps the search direction with the gradient is incorporated as in the unconstrained case. Thus, a descent direction d_k is computed as an approximate solution of the subproblem (3.21) that lies within the projected subspace

$$\mathcal{G}_k \triangleq \operatorname{span}\left\{\mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} p_l, \cdots, \mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} p_{k-1}, \mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\right\}, \quad \text{with } \; l = \max\{0, k-m+1\}.$$

The search direction p_k is then determined by modifying d_k as in Steps 7–8 of Algorithm 3. In the following discussion, let $\bar{B}_k$ denote a matrix whose columns form a basis of $\bar{\mathcal{G}}_k$. Similarly, let B_k denote a matrix whose columns form a basis of $\mathcal{G}_k$, and an orthonormal basis Z_k is defined in terms of the economy-size QR decomposition $B_k = Z_k T_k$, with T_k a nonsingular upper-triangular matrix.

The vector d_k is computed as $d_k = Z_k q$, where q is the solution of the symmetric positive-definite equations

$$Z_k^{\mathrm{T}} H_k Z_k q = -Z_k^{\mathrm{T}} \nabla f(x_k).$$

Note that B_k is the matrix $\bar{B}_k$ with zeros in the rows corresponding to indices in the current working set. The search direction may be computed efficiently by using a Cholesky factor of the "projected" reduced Hessian matrix $R_k^{\mathrm{T}} R_k = Z_k^{\mathrm{T}} H_k Z_k$.

Once d_k has been computed, the search direction p_k is derived as described in Algorithm 3, and then the next iterate x_{k+1} is found using the quasi-Wolfe search described in Section 3.2. The associated working set $\mathcal{W}_k$ is then updated and the projected matrix factors B_k, Z_k, and T_k are modified to reflect the changes in the working set. If the projected gradient at x_{k+1} contains components outside of range(Z_k), then it can be added to the basis. If the value of the scalar $\rho = \|\big(I - Z_k Z_k^{\mathrm{T}}\big)\nabla f(x_{k+1})\|$ is zero, then the new gradient lies in $\mathcal{G}_k$ and $\nabla f(x_{k+1})$ is *rejected* for inclusion in $\bar{\mathcal{G}}_k$. In this case, no further updates to the factors of B_k are needed. Otherwise, the dimension of $\mathcal{G}$ increases by one and the gradient $\nabla f(x_{k+1})$ is *accepted.* In this case, $\bar{B}_k$ gains a new column and the change must be incorporated in the QR factors of B_k analogous to (3.18). The matrix updates associated with changes in the working set and the basis are based on the work of Daniel et al. [19] and are omitted here; for a detailed description, see [30, Section 5].

The convergence of a projected-search active-set method has been established in Theorem 3.3.2. Moreover, if the eigenvalues of the projected approximate Hessian are uniformly bounded, then the projected gradient converges to zero as shown in the theorem below.

Theorem 3.4.1. *Let* $\{\, x_k \,\}$ *be a sequence of iterates generated by* Algorithm UBOPT. *In addition to assumptions of* Theorem 3.3.2*, if there exist a constants* γ *such that every eigenvalue of the projected approximate Hessian satisfies*

$$0 < \lambda(\mathit{\Pi}_k^{\mathrm{T}} H_k \mathit{\Pi}_k) \le \gamma < \infty$$

for all k*, where* $\mathit{\Pi}_k$ *is a matrix with orthonormal columns that spans the set of projected directions with respect to the working set* $\mathcal{W}_k(x_k)$*, then*

$$\lim_{k\to\infty} \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\| = 0.$$

Proof. Let d_k denote the approximate solution to the subproblem (3.21) within the subspace spanned by columns of $\mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} B_k$, and let Z_k be the orthogonal factor of the thin QR decomposition of $\mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} B_k$. Then

$$|\nabla f(x_k)^{\mathrm{T}} d_k| = |\nabla f(x_k)^{\mathrm{T}} Z_k (Z_k^{\mathrm{T}} H_k Z_k)^{-1} Z_k^{\mathrm{T}} \nabla f(x_k)| \ge \|Z_k^{\mathrm{T}} \nabla f(x_k)\|^2 / \lambda_{\max}(\mathit{\Pi}_k^{\mathrm{T}} H_k \mathit{\Pi}_k),$$

for all k, where $\lambda_{\max}(\mathit{\Pi}_k^{\mathrm{T}} H_k \mathit{\Pi}_k)$ represents the largest eigenvalue of the projected approximate Hessian. As represents the largest eigenvalue of the projected approximate Hessian. As $\mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)$ lies in the column space of Z_k,

$$\|Z_k^{\mathrm{T}} \nabla f(x_k)\| = \|Z_k^{\mathrm{T}} \mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\| = \|\mathit{\Pi}_k \mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\| = \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\|.$$

It follows that

$$|\nabla f(x_k)^{\mathrm{T}} d_k| \ge \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\|^2 / \lambda_{\max}(\mathit{\Pi}_k^{\mathrm{T}} H_k \mathit{\Pi}_k) \ge \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\|^2 / \gamma.$$

Then

$$0 = \lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} p_k| \ge \lim_{k\to\infty} |\nabla f(x_k)^{\mathrm{T}} d_k| \ge \lim_{k\to\infty} \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\|^2 / \gamma.$$

Therefore,

$$\lim_{k\to\infty} \|\mathit{\Pi}_k^{\mathrm{T}} \nabla f(x_k)\| = 0.$$

□

Furthermore, it is stated in Theorem 3.3.3 that, if the sequence of iterates $\{\, x_k \,\}$ converges to a nondegenrate stationary point, then the optimal active set can be identified with a finite number of iterations, i.e., UBOPT will eventually reduce to the L-RHR for the unconstrained minimization with respect to the inactive variables. Therefore, Algorithm UBOPT eventually has the same convergence properties as L-RHR.

3.5 Projected-Search Interior Methods

3.5.1 The general framework

A typical interior method for problem (BC) is based on minimizing a sequence of unconstrained functions $M(v\,;\mu)$ parameterized by a positive scalar μ. In general, the function $M(v\,;\mu)$ is either not defined or unbounded for some values of the variables, which implies that the v are subject to implicit bound constraints during the minimization. It follows that for a given μ, the problem to be solved has the general form

$$\underset{v\in\mathbb{R}^{n_v}}{\text{minimize}} \;\; M(v\,;\mu) \quad \text{subject to} \;\; v \in \Omega, \tag{IPBC}$$

where $\Omega = \{\, x \in \mathbb{R}^n : \ell_v < v < u_v \,\}$, with ℓ_v and u_v fixed n_v-vectors of lower and upper bounds on v. Such a function $M(v\,;\mu)$ typically has the property that if v approaches the boundary of Ω, the value of the function becomes $+\infty$ (see Section 2.2). For example, in the case of the classical logarithmic barrier method, the function $M(v\,;\mu)$ is given by

$$f(x) - \sum_{j=1}^{n} \mu \ln\big(x_j - \ell_j\big) - \sum_{j=1}^{n} \mu \ln\big(u_j - x_j\big),$$

so that $v = x$ and the implicit bounds are $\ell < x < u$.

The proposed projected-search line-search method for problem (IPBC) generates a sequence of feasible iterates $\{\, v_k \,\}_{k=0}^{\infty}$ such that $v_{k+1} = \mathbf{proj}_{\,\Omega_k}(v_k + \alpha_k \Delta v_k)$, where Δv_k is a descent direction for $M(v\,;\mu)$, α_k is a quasi-Wolfe step, and $\mathbf{proj}_{\,\Omega_k}(v)$ is the projection of v onto the perturbed feasible region

$$\Omega_k = \{\, v : v_k - \sigma(v_k - \ell_v) \le v \le v_k + \sigma(u_v - v_k) \,\}, \tag{3.23}$$

where σ a fixed positive scalar such that $0 < \sigma < 1$. The quantity σ may be interpreted as the "fraction to the boundary" parameter used in many conventional interior methods.

It is worth mentioning that, although this chapter concerns bound-constrained problems only, the framework of projected-search interior methods proposed here may also be applied to general nonlinearly constrained problems given an appropriate merit function $M(v\,;\mu)$. Projected-search interior methods have the potential of requiring fewer iterations than a conventional interior method, thereby reducing the number of times that a search direction need be computed. Section 5.2 gives numerical results for a primal-dual projected-search interior method based on the method of Forsgren and Gill [34].

3.5.2 Convergence analysis

In this section, the convergence properties of the class of projected-search interior methods are established. As the parameter μ is fixed in problem (IPBC), the notation can be simplified by writing $M(v) = M(v\,;\mu)$. The projected search is performed on the univariate function $\psi_k(\alpha) = M\big(\mathbf{proj}_{\varOmega_k}(v_k + \alpha_k \varDelta v_k)\big)$.

Theorem 3.5.1 (Interior projected search with a quasi-Armijo search)**.** *Let $M(v)$ be a continuously differentiable function defined on $\varOmega$ such that $M(v) \to \infty$ as v approaches the boundary of $\varOmega$. Assume that $v_0 \in \varOmega$ is chosen such that the level set $\mathcal{L}\big(M(v_0)\big)$ is bounded, and $\{\, v_k \,\}$ is defined by $v_{k+1} = \mathbf{proj}_{\varOmega_k}(v_k + \alpha_k \varDelta v_k)$, where $\mathbf{proj}_{\varOmega_k}(v)$ is the projection of v onto the set*

$$\varOmega_k = \big\{\, v : v_k - \sigma(v_k - \ell_v) \le v \le v_k + \sigma(u_v - v_k) \,\big\},$$

with σ a fixed positive scalar such that $0 < \sigma < 1$, $\varDelta v_k$ is a descent direction, and α_k is a quasi-Armijo step. Also assume that $\|\varDelta v_k\| \le \theta$ for some constant θ independent of k Then

$$\lim_{k\to\infty} |\nabla M(v_k)^{\mathrm{T}} \varDelta v_k| = 0.$$

Proof. Observe that the quasi-Armijo condition implies that $\{\, M(v_k) \,\}$ is a strictly decreasing sequence. As the set $\mathcal{L}\big(M(v_0)\big)$ is bounded, it follows that $\{\, M(v_k) \,\}$ converges, with

$$0 = \lim_{k\to\infty} M(v_k) - M(v_{k+1}) \ge \lim_{k\to\infty} \alpha_k \eta_A |\nabla M(v_k)^{\mathrm{T}} \varDelta v_k| = 0.$$

The proof is by contradiction. Suppose that $|\nabla M(v_k)^{\mathrm{T}} \varDelta v_k| \not\to 0$ as $k \to \infty$, then there must exist some $\bar{\epsilon} > 0$ such that $|\nabla M(v_k)^{\mathrm{T}} \varDelta v_k| > \bar{\epsilon}$ infinitely often. Let $\mathcal{G} = \big\{\, k : |\nabla M(v_k)^{\mathrm{T}} \varDelta v_k| > \bar{\epsilon} \,\big\}$, then $\alpha_k \to 0$ for $k \in \mathcal{G}$. Let $\gamma = \inf_{v \in \mathcal{L}(M(v_0)),i} \{\, v_i - \ell_i, u_i - v_i \,\}$. By the continuity of $M(v)$ in $\varOmega$, the level set $\mathcal{L}(M(v_0))$ is closed. Hence, $\mathcal{L}(M(v_0))$ is a compact subset of the open set $\varOmega$, which implies that $\gamma > 0$. By the quasi-Armijo condition, each v_k lies in $\mathcal{L}(M(v_0))$. Therefore, for each i, $u_i - [v_k]_i > \gamma$, and $[v_k]_i - l_i > \gamma$ for all k. For each k, define the step $\beta_k = \alpha_k/\sigma$. As $\{\, \|\varDelta v_k\| \,\}$ is uniformly bounded by θ, there exists $\bar{k}$ such that each componet of $\beta_k \varDelta v_k$ satisfies $|[\, \beta_k \varDelta v_k \,]_i| < \sigma\gamma$ for all $k \geq \bar{k}$ in $\mathcal{G}$. It follows that $v_k + \beta_k \varDelta v_k \in \varOmega_k$, which implies $\mathbf{proj}_{\,\varOmega_k}(v_k + \beta_k \varDelta v_k) = v_k + \beta_k \varDelta v_k$.

Let $\bar{\mathcal{G}}$ denote the indices $k \geq \bar{k}$ of iterations at which a reduction in the initial step length was necessary, i.e., $\bar{\mathcal{G}} = \{\, k : t_k > 0,\ k \in \mathcal{G},\ k \geq \bar{k} \,\}$. Since α_k converges to zero, $\bar{\mathcal{G}}$ must be an infinite set. By definition of the quasi-Armijo step,

$$M(v_k + \beta_k \varDelta v_k) = M(\mathbf{proj}_{\,\varOmega_k}(v_k + \beta_k \varDelta v_k)) > M(v_k) + \beta_k \eta_A \nabla M(v_k)^{\mathrm{T}} \varDelta v_k, \ \text{ for all } \ k \in \bar{\mathcal{G}}.$$

Adding $-\beta_k \nabla M(v_k)^{\mathrm{T}} \varDelta v_k$ to both sides and rearranging gives

$$\begin{aligned} M(v_k + \beta_k \varDelta v_k) - M(v_k) - \beta_k \nabla M(v_k)^{\mathrm{T}} \varDelta v_k &> -\beta_k (1 - \eta_A) \nabla M(v_k)^{\mathrm{T}} \varDelta v_k \\ &> \beta_k (1 - \eta_A) \bar{\epsilon}, \ \text{ for all } \ k \in \bar{\mathcal{G}}. \end{aligned} \tag{3.24}$$

The Taylor expansion of $M(v_k + \beta_k \varDelta v_k)$ gives

$$M(v_k + \beta_k \varDelta v_k) - M(v_k) - \beta_k \nabla M(v_k)^{\mathrm{T}} \varDelta v_k = \beta_k \int_0^1 \big(\nabla M(v_k + \tau \beta_k \varDelta v_k) - \nabla M(v_k)\big)^{\mathrm{T}} \varDelta v_k \, d\tau. \tag{3.25}$$

If $\|\cdot\|_D$ denotes the norm dual to $\|\cdot\|$, i.e., $\|v\|_D = \max_{v \neq 0} |v^{\mathrm{T}} v|/\|v\|$, then

$$\big|\big(\nabla M(v_k + \tau \beta_k \varDelta v_k) - \nabla M(v_k)\big)^{\mathrm{T}} \varDelta v_k\big| \leq \|\nabla M(v_k + \tau \beta_k \varDelta v_k) - \nabla M(v_k)\|_D \|\varDelta v_k\|.$$

If this inequality is substituted in (3.25), it then follows from (3.24) that

$$\begin{aligned} (1 - \eta_A)\bar{\epsilon} &< \int_0^1 \big(\nabla M(v_k + \tau \beta_k \varDelta v_k) - \nabla M(v_k)\big)^{\mathrm{T}} \varDelta v_k \, d\tau \\ &\leq \max_{0 \leq \tau \leq 1} \|\nabla M(v_k + \tau \beta_k \varDelta v_k) - \nabla M(v_k)\|_D \|\varDelta v_k\|, \ \text{ for all } \ k \in \bar{\mathcal{G}}. \end{aligned}$$

The continuity of ∇M implies that there exists some $\tau_k \in [0, \beta_k]$ such that

$$\max_{0 \leq \tau \leq 1} \|\nabla M(v_k + \tau \beta_k \varDelta v_k) - \nabla M(v_k)\|_D = \|\nabla M(v_k + \tau_k \varDelta v_k) - \nabla M(v_k)\|_D.$$

Then

$$(1 - \eta_A)\bar{\epsilon} < \|\nabla M(v_k + \tau_k \varDelta v_k) - \nabla M(v_k)\|_D \|\varDelta v_k\|. \tag{3.26}$$

However, $\alpha_k \Delta v_k \to 0$ implies $\tau_k \Delta v_k \to 0$ for $k \in \mathcal{G}$, and the continuity of ∇M gives

$$\|\nabla M(v_k + \tau_k \Delta v_k) - \nabla M(v_k)\|_D \to 0.$$

As $\{\, \|\Delta v_k\| \,\}$ is uniformly bounded above by θ, the right-hand side of (3.26) converges to zero, which gives the required contradiction.

□

Theorem 3.5.2 (Interior projected search with a quasi-Wolfe search)**.** *Let $M(v)$ be a continuously differentiable function defined on Ω such that $M(v) \to \infty$ as v approaches the boundary of Ω. Assume that $v_0 \in \Omega$ is chosen such that the level set $\mathcal{L}\big(M(v_0)\big)$ is bounded, and $\{\, v_k \,\}$ is defined by $v_{k+1} = \mathbf{proj}_{\,\Omega_k}(v_k + \alpha_k \Delta v_k)$, where $\mathbf{proj}_{\,\Omega_k}(v)$ is the projection of v onto the set*

$$\Omega_k = \big\{\, v : v_k - \sigma(v_k - \ell_v) \le v \le v_k + \sigma(u_v - v_k) \,\big\},$$

with σ a fixed positive scalar such that $0 < \sigma < 1$, Δv_k is a descent direction, and α_k is a quasi-Wolfe step. Also assume that $\|\Delta v_k\| \le \theta$ for some constant θ independent of k Then

$$\lim_{k\to\infty} |\nabla M(v_k)^{\mathrm{T}} \Delta v_k| = 0.$$

Proof. The first quasi-Wolfe condition ($\mathbf{C}_1$) is equivalent to the quasi-Armijo condition, and the arguments in the proof of Theorem 3.5.1 may be used to show that $\{\, M(v_k) \,\}$ is a convergent sequence. This implies that

$$\lim_{k\to\infty} \alpha_k \nabla M(v_k)^{\mathrm{T}} \Delta v_k = 0.$$

The proof is by contradiction. Suppose that $|\nabla M(v_k)^{\mathrm{T}} \Delta v_k| \not\to 0$ as $k \to \infty$, then there exists some $\bar\epsilon > 0$ such that $|\nabla M(v_k)^{\mathrm{T}} \Delta v_k| > \bar\epsilon$ infinitely often. Let $\mathcal{G} = \big\{\, k : |\nabla M(v_k)^{\mathrm{T}} \Delta v_k| > \bar\epsilon \,\big\}$, then it must be that $\alpha_k \to 0$ for $k \in \mathcal{G}$. As $\{\, \|\Delta v_k\| \,\}$ is uniformly bounded above by θ, $\alpha_k \Delta v_k \to 0$ for $k \in \mathcal{G}$.

If the quasi-Wolfe condition ($\mathbf{C}_2$) is satisfied, then

$$\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(\Delta v_k) \ge -\eta_W |\nabla M(v_k)^{\mathrm{T}} \Delta v_k|.$$

Similarly, if the quasi-Wolfe condition ($\mathbf{C}_4$) is satisfied, then

$$\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(\Delta v_k) \ge 0 \ge -\eta_W |\nabla M(v_k)^{\mathrm{T}} \Delta v_k|.$$

In either case, as $\nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k < 0$, it must hold that

$$\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k \geq (1-\eta_W)|\nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k| > (1-\eta_W)\bar{\epsilon}$$

for $k \in \mathcal{G}$. The application of the triangle inequality yields

$$\begin{aligned} 0 < (1-\eta_W)\bar{\epsilon} &< \left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k\right| \\ &\leq \left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k)\right| \\ &\qquad + \left|\nabla M(v_k)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k\right|. \end{aligned} \tag{3.27}$$

Let $\|\cdot\|_D$ denote the norm dual to $\|\cdot\|$, then

$$\begin{aligned} &\left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(v_k) - \nabla M(v_k)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k)\right| \\ &\qquad \leq \|\nabla M\big(v_k(\alpha_k)\big) - \nabla M(v_k)\|_D \|P_{v_k(\alpha_k)}(\mathit{\Delta} v_k)\| \leq \|\nabla M\big(v_k(\alpha_k)\big) - \nabla M(v_k)\|_D \|\mathit{\Delta} v_k\|. \end{aligned}$$

As ∇M is continuous and $\|\mathit{\Delta} v_k\|$ is uniformly bounded, the right-hand side of this inequality must converge to zero for $k \in \mathcal{G}$, which implies that

$$\left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k)\right| \to 0, \ \text{ for } k \in \mathcal{G}.$$

Basic norm inequalities give

$$\begin{aligned} \left|\nabla M(v_k)^{\mathrm{T}} P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k\right| &\leq \|\nabla M(v_k)\|_D \|P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \mathit{\Delta} v_k\| \\ &= \|\nabla M(v_k)\|_D \|P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - P_{v_k}(\mathit{\Delta} v_k)\|. \end{aligned}$$

Let $\gamma = \inf_{v \in \mathcal{L}(M(v_0)), i} \big\{ v_i - \ell_i, u_i - v_i \big\}$. By the same arguments in the proof of Theorem 3.5.1, $\gamma > 0$, and for each i, $u_i - [v_k]_i > \gamma$, and $[v_k]_i - l_i > \gamma$ for all k. Therefore, for $k \in \mathcal{G}$ sufficiently large such that each component $[\,\alpha_k \mathit{\Delta} v_k\,]_i < \sigma\gamma$,

$$P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) = P_{v_k}(\mathit{\Delta} v_k) = \mathit{\Delta} v_k.$$

Therefore,

$$\|\nabla M(v_k)\|_D \|P_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - P_{v_k}(\mathit{\Delta} v_k)\| \to 0, \ \text{ for } k \in \mathcal{G},$$

and consequently

$$\left|\nabla M(v_k)^{\mathrm{T}} P_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} \varDelta v_k\right| \to 0, \quad \text{for } k \in \mathcal{G}.$$

It follows that the right-hand side of (3.27) converges to zero for $k \in \mathcal{G}$, which gives the required contradiction.

It remains to consider the case where the quasi-Wolfe condition ($\mathbf{C}_3$) is satisfied, i.e.,

$$\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) \geq -\eta_W |\nabla M(v_k)^{\mathrm{T}} \varDelta v_k|.$$

The assumption that $\nabla M(v_k)^{\mathrm{T}} \varDelta v_k < 0$ gives

$$\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} \varDelta v_k \geq (1-\eta_W)|\nabla M(v_k)^{\mathrm{T}} \varDelta v_k| > (1-\eta_W)\bar{\epsilon}$$

for $k \in \mathcal{G}$, which implies that

$$\begin{aligned} 0 < (1-\eta_W)\bar{\epsilon} &< \left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} \varDelta v_k\right| \\ &\leq \left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k)\right| \\ &\qquad + \left|\nabla M(v_k)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} \varDelta v_k\right|. \end{aligned} \tag{3.28}$$

The definition of the dual norm yields

$$\begin{aligned} &\left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k)\right| \\ &\qquad \leq \|\nabla M\big(\varDelta v_k(\alpha_k)\big) - \nabla M(v_k)\|_D \|P^{-}_{v_k(\alpha_k)}(\varDelta v_k)\| \leq \|\nabla M\big(v_k(\alpha_k)\big) - \nabla M(v_k)\|_D \|\varDelta v_k\|. \end{aligned}$$

From the continuity of ∇M and uniform boundedness of $\|\varDelta v_k\|$, the right-hand side of the above inequality converges to zero for $k \in \mathcal{G}$, which means that

$$\left|\nabla M\big(v_k(\alpha_k)\big)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k)\right| \to 0, \quad \text{for } k \in \mathcal{G}.$$

Also,

$$\begin{aligned} \left|\nabla M(v_k)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \nabla M(v_k)^{\mathrm{T}} \varDelta v_k\right| &\leq \|\nabla M(v_k)\|_D \|P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - \varDelta v_k\| \\ &= \|\nabla M(v_k)\|_D \|P^{-}_{v_k(\alpha_k)}(\varDelta v_k) - P_{v_k}(\varDelta v_k)\|. \end{aligned}$$

As $\|v_k(\alpha_k) - v_k\| \le \|\alpha_k \mathit{\Delta} v_k\| \to 0$ for $k \in \mathcal{G}$, arguments analogous to those used to establish convergence in cases $(\mathbf{C}_2)$ and $(\mathbf{C}_4)$ give

$$P^{-}_{v_k(\alpha_k)}(\mathit{\Delta} v_k) = P_{v_k}(\mathit{\Delta} v_k) = \mathit{\Delta} v_k \text{ for } k \in \mathcal{G} \text{ sufficiently large,}$$

in which case

$$\|\nabla M(v_k)\|_D \|P^{-}_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - P_{v_k}(\mathit{\Delta} v_k)\| \to 0, \text{ for } k \in \mathcal{G}.$$

This implies that

$$\left|\nabla M(v_k)^{\mathrm{T}} P^{-}_{v_k(\alpha_k)}(\mathit{\Delta} v_k) - \nabla M(v_k)^{\mathrm{T}} \mathit{\Delta} v_k\right| \to 0, \text{ for } k \in \mathcal{G}.$$

It follows that the right-hand side of (3.28) converges to zero for $k \in \mathcal{G}$, which gives the required contradiction. □

Chapter 3, as well as the numerical results in Sections 5.1–5.2, is partially a reprint of the paper "Projected-search methods for bound-constrained optimization" by Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang, available on arXiv:2110.08359 [math.OC]. Manuscript submitted for publication, 2021. The dissertation author served as the primary investigator and author of the paper. Additionally, Section 3.4 partially reprints the paper by Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang, titled "A limited-memory reduced-Hessian method for bound-constrained optimization." Center for Computational Mathematics Report CCoM 21-01, Center for Computational Mathematics, University of California San Diego, La Jolla, CA, 2021. The dissertation author was the primary author of the content.

Chapter 4

A Projected-Search Interior Method for Nonlinear Optimization

4.1 Introduction

This chapter concerns the formulation and analysis of a new primal-dual interior method for solving nonlinear optimization problems of the form

$$\underset{x\in\mathbb{R}^n, s\in\mathbb{R}^m}{\text{minimize}} \;\; f(x) \quad \text{subject to} \;\; c(x) - s = 0, \quad s \geq 0, \tag{NIPs}$$

where $c : \mathbb{R}^n \mapsto \mathbb{R}^m$ and $f : \mathbb{R}^n \mapsto \mathbb{R}$ are twice-continuously differentiable. (The slack variables s serve to convert the inequalities $c(x) \geq 0$ into a mixture of equalities and inequalities that do not require the need to know an initial point for which c is strictly positive.)

In [46], Gill, Kungurtsev and Robinson propose an algorithm for (NIPs) based on using a shifted primal-dual penalty-barrier function as a merit function for a primal-dual path-following method. This function involves a primal-dual shifted penalty term for the equality constraints $c(x) - s = 0$ and an analogous primal-dual shifted barrier term for the inequalities $s \geq 0$. It is shown that a specific approximate Newton method for the unconstrained minimization of the merit function generates search directions that are identical to those associated with a variant of the conventional path-following method in which the perturbation of the complementarity condition does not need to go to zero.

The proposed method is based on a newly formulated merit function that includes shifts for the dual variables as well as the slack variables s. (For problems with a mixture of upper and lower bounds on x and s, the method may be regarded as shifting both the primal and dual variables, see Appendix A.) Shifts on the dual variables allow the method to be safely "warm started" from a good approximate solution and eliminates the ill-conditioning of the associated linear equations that may occur when the dual variables are close to zero.

The shifted primal-dual penalty-barrier function includes logarithmic barrier terms that create a singularity at the boundary of the primal-dual shifted feasible region, which implies that the variables are subject to implicit bound constraints during the minimization. A novel projected-search method is designed for the minimization of the all-shifted penalty-barrier function, which employs a flexible non-monotone *quasi-Armijo* line search. Unlike conventional interior methods, projected-search interior methods project the underlying search direction onto a subset of the feasible region defined by perturbing the bounds. With this approach the direction of the search path may change multiple times along the boundary of the perturbed feasible region at the cost of computing a single direction. Projected-search interior methods have the potential of requiring fewer iterations than a conventional interior method, thereby reducing the number of times that a search direction must be computed.

The projected-search method generates a sequence of feasible iterates $\{v_k\}_{k=0}^{\infty}$ such that $v_{k+1} = \mathbf{proj}_{\mathit{\Omega}_k}(v_k + \alpha_k \mathit{\Delta} v_k)$, where $\mathbf{proj}_{\mathit{\Omega}_k}(v)$ is the projection of the vector v of primal-dual variables onto a perturbed feasible region $\mathit{\Omega}_k$. Under mild assumptions, it is shown that there exists a limit point of the computed iterates that is either an infeasible stationary point, or a complementary approximate Karush-Kuhn-Tucker point (KKT), i.e., it satisfies reasonable stopping criteria and is a KKT point under a complementary approximate KKT regularity condition that is weaker than MFCQ (see Andreani, Martínez, Ramos and Svaiter [1]).

The rest of the chapter is organized as follows. Section 4.2 reviews the method of Gill, Kungurtsev and Robinson [46], which is based on minimizing a shifted primal-dual penalty-barrier function. In the neighborhood of a solution, under suitable assumptions, the method is equivalent to a variant of the primal-dual path-following method in which the slack variables are shifted. In Section 4.3 this method is extended to include shifts on the dual variables as well as the slacks to formulate an all-shifted primal-dual penalty-barrier function. In Section 4.4, a projected-search algorithm is proposed for minimizing the all-shifted primal-dual penalty-barrier function for fixed penalty and barrier parameters. The convergence of this algorithm is established under certain assumptions. Section 4.5 presents an algorithm for solving problem (NIPs) that builds upon the work from Section 4.4. Global convergence results of the algorithm are also established.

4.2 Background: A Primal-Dual Method Based on Shifting the Slacks

Given an appropriate constraint qualification, the first-order optimality conditions for problem (NIPs) are given by

$$\left.\begin{aligned} \nabla f(x) - J(x)^{\mathrm{T}} y = 0, &\qquad y - w = 0, \\ c(x) - s = 0, &\qquad s \geq 0, \\ s \cdot w = 0, &\qquad w \geq 0, \end{aligned}\right\} \tag{4.1}$$

where the vectors y and w constitute the Lagrange multipliers for the equality constraint $c(x) - s = 0$ and nonnegativity constraint $s \geq 0$ respectively (see Theorem 2.2.7). Any point satisfying the conditions (4.1) is called a first-order KKT point.

Primal-dual path-following methods generate a sequence of iterates that approximate a continuous primal-dual path that passes through a solution of (NIPs). Points on this path satisfy a system of nonlinear equations that represent the deviations from a perturbation of the first-order optimality conditions (4.1). In a conventional path-following approach, the perturbed optimality conditions correspond to replacing the equality constraints and complementarity conditions of (4.1) by $c(x) - s = \mu y$ and $s \cdot w = \mu e$, where μ is a small positive parameter such that $\mu \to 0$. This method is closely related to penalty-barrier methods for solving (NIPs). Under certain conditions on f and c the continuous trajectory of penalty-barrier minimizers associated with a continuous penalty-barrier parameter μ coincides with the primal-dual path.

In the neighborhood of a first-order KKT point, computing the search direction as the solution of the Newton equations for a zero of the perturbed optimality conditions provides the favorable local convergence rate associated with Newton's method. Given the close connection with penalty-barrier methods, solving the Newton equations provides an alternative to solving the ill-conditioned equations associated with a conventional penalty-barrier method. In this context, the penalty-barrier function may be regarded as a merit function for forcing convergence of the sequence of Newton iterates of the path-following method. For examples of this approach, see Byrd, Hribar and Nocedal [12], Wächter and Biegler [83], Forsgren and Gill [34], and Gertz and Gill [44].

When implemented with exact second derivatives, path-following interior methods often converge in few iterations–even for very large problems. As the dimension and zero/nonzero structure of the Jacobian matrix remains fixed, the Newton equations may be solved efficiently using advanced "off-the-shelf" linear algebra software. On the negative side, although conventional path-following interior methods are very effective for solving "one-off" problems, they are difficult to adapt to solving a sequence of related problems using so-called "warm starts", i.e., using the solution of one problem as an initial estimate of the solution of the next.

In a conventional path-following interior method, it is necessary to force $\mu \to 0$ to ensure that points near the path eventually satisfy the optimality conditions (4.1). However, if an augmented Lagrangian method defined with multiplier estimate y^E and penalty parameter μ^P is used to minimize $f(x)$ subject to $c(x) = 0$, then perturbed conditions of the form $c(x) = \mu^P(y^E - y)$ hold at a minimizer. It follows that μ^P need not go to zero if y^E is chosen converge to the optimal multipliers. Motivated by this observation, the method of Gill, Kungurtsev and Robinson [46] is based on the perturbed optimality conditions

$$\left.\begin{aligned} \nabla f(x) - J(x)^{\mathrm{T}} y = 0, & \qquad & y - w = 0, \\ c(x) - s = \mu^P(y^E - y), & \qquad & s \ge 0, \\ s \cdot w = \mu^B(w^E - w), & \qquad & w \ge 0, \end{aligned}\right\} \tag{4.2}$$

where μ^P and μ^B are positive scalars and y^E and w^E denote estimates of the Lagrange multipliers for the constraints $c(x) - s = 0$ and $s \ge 0$, respectively. The perturbed complementarity condition in (4.2) may be written in the form $(s + \mu^B e) \cdot w = \mu^B w^E$, which implies that if $w^E > 0$ then $s + \mu^B e > 0$ and $w > 0$. Gill, Kungurtsev and Robinson show that an appropriate merit function for a path-following interior method based on the conditions (4.2) is the shifted primal-dual penalty-barrier function

$$\begin{aligned} M(x, s, y, w\,; y^E, w^E, \mu^P, \mu^B) = f(x) &- (c(x) - s)^{\mathrm{T}} y^E \\ &+ \frac{1}{2\mu^P}\|c(x) - s\|^2 + \frac{1}{2\mu^P}\|c(x) - s + \mu^P(y - y^E)\|^2 \\ &- \sum_{i=1}^{m} \mu^B w_i^E \ln\big(s_i + \mu^B\big) - \sum_{i=1}^{m} \mu^B w_i^E \ln\big(w_i(s_i + \mu^B)\big) + \sum_{i=1}^{m} w_i(s_i + \mu^B). \end{aligned}$$

In the neighborhood of a minimizer of (NIPs) satisfying certain second-order optimality conditions, the Newton equations for a zero of the conditions (4.2) are equivalent to the Newton equations for a minimizer of M. Under certain assumptions, a limit point of the iterates generated by the algorithm may always be found that is either an infeasible stationary point or a complementary approximate KKT point (see Andreani, Martínez and Svaiter [2]). The reader is referred to Gill, Kungurtsev and Robinson [46] for more details.

In the following section, the Gill, Kungurtsev and Robinson algorithm is extended to include shifts on the dual variables w as well as the slack variables s.

4.3 An All-Shifted Primal-Dual Penalty-Barrier Function

To include shifts for the dual variables, the following perturbed optimality conditions are considered:

$$\left.\begin{aligned} \nabla f(x) - J(x)^{\mathrm{T}} y &= 0, & y - w &= 0, \\ c(x) - s &= \mu^{P}(y^{E} - y), & s &\geq 0, \\ s \cdot w &= \mu^{B}(w^{E} - w) + \mu^{B}(s^{E} - s), & w &\geq 0, \end{aligned}\right\} \tag{4.3}$$

where $y^{E} \in \mathbb{R}^{m}$ is an estimate of a Lagrange multiplier vector for the constraint $c(x) - s = 0$, $w^{E} \in \mathbb{R}^{m}$ is an estimate of a Lagrange multiplier for the constraint $s \geq 0$, $s^{E} \in \mathbb{R}^{m}$ is an estimate of the optimal slacks, and μ^{P} and μ^{B} are positive scalars. The last equation of (4.3) may be written in the form $(s + \mu^{B} e) \cdot (w + \mu^{B} e) = \mu^{B}(s^{E} + w^{E} + \mu^{B} e)$, which implies that if $s^{E} + w^{E} + \mu^{B} e > 0$ then $s + \mu^{B} e > 0$ and $w + \mu^{B} e > 0$. If $F(x,\, s,\, y,\, w;\, s^{E},\, y^{E},\, w^{E},\, \mu^{P},\, \mu^{B})$ denotes the function

$$F(x, s, y, w\,; s^{E}, y^{E}, w^{E}, \mu^{P}, \mu^{B}) = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y \\ y - w \\ c(x) - s + \mu^{P}(y - y^{E}) \\ s \cdot w - \mu^{B}(w^{E} - w + s^{E} - s) \end{pmatrix}, \tag{4.4}$$

then any point $(x,\, s,\, y,\, w)$ that satisfies the perturbed optimality conditions (4.3) must satisfy $F(x,\, s,\, y,\, w\ ;\ s^{E},\, y^{E},\, w^{E},\, \mu^{P},\, \mu^{B}) = 0$. Let $F(v)$ denote the function at a given point $v = (x, s, y, w)$. The Newton equations for the step Δv are given by $F'(v)\Delta v = -F(v)$, i.e.,

$$\begin{pmatrix} H(x,y) & 0 & -J(x)^{\mathrm{T}} & 0 \\ 0 & 0 & I_m & -I_m \\ J(x) & -I_m & \mu^{P} I_m & 0 \\ 0 & W + \mu^{B} I_m & 0 & S + \mu^{B} I_m \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta s \\ \Delta y \\ \Delta w \end{pmatrix} = - \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y \\ y - w \\ c(x) - s + \mu^{P}(y - y^{E}) \\ s \cdot w - \mu^{B}(w^{E} - w + s^{E} - s) \end{pmatrix}, \tag{4.5}$$

where S and W denote diagonal matrices with diagonal entries s_i and w_i such that $s_i + \mu^{B} > 0$ and $w_i + \mu^{B} > 0$.

Next, a penalty-barrier function M needs to be formulated such that in a neighborhood of a minimizer of M, the Newton equations for minimizing M approximate the Newton equations

(4.5). such that in a neighborhood of a minimizer of M, the Newton equations for minimizing M approximate the Newton equations (4.5). Consider the shifted primal-dual penalty-barrier function

$$\left.\begin{aligned} M(x,s,y,w\,;s^E,y^E,w^E,\mu^P,\mu^B) = {}& \underbrace{f(x)}_{(A)} \underbrace{-(c(x)-s)^{\mathrm{T}} y^E}_{(B)} + \underbrace{\frac{1}{2\mu^P}\|c(x)-s\|^2}_{(C)} \\ &+ \underbrace{\frac{1}{2\mu^P}\|c(x)-s+\mu^P(y-y^E)\|^2}_{(D)} \\ & \underbrace{-2\sum_{i=1}^{m} \mu^B(w_i^E+s_i^E+\mu^B)\ln(s_i+\mu^B)}_{(E)} \\ & \underbrace{-\sum_{i=1}^{m} \mu^B(w_i^E+s_i^E+\mu^B)\ln(w_i+\mu^B)}_{(F)} \\ &+ \underbrace{\sum_{i=1}^{m} w_i(s_i+\mu^B)}_{(G)} + \underbrace{2\mu^B\sum_{i=1}^{m} s_i}_{(H)}\,. \end{aligned}\right\} \tag{4.6}$$

Let S^E denote the diagonal matrix with diagonal entries s_i^E. Similarly, let

$$S_B = S + \mu^B I_m, \quad S_B^E = S^E + \mu^B I_m \text{ and } W_B = W + \mu^B I_m.$$

Given the positive-definite matrices

$$D_P = \mu^P I_m \quad \text{and} \quad D_B = S_B W_B^{-1},$$

and auxiliary vectors

$$\pi^Y(x) = y^E - \frac{1}{\mu^P}(c(x)-s) \quad \text{and} \quad \pi^W(s) = \mu^B(S+\mu^B I)^{-1}(w^E - s + s^E),$$

the gradient ∇M may be written as

$$\nabla M = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}}\big(\pi^Y + (\pi^Y - y)\big) \\ (\pi^Y - y) + (\pi^Y - \pi^W) + (w - \pi^W) \\ -D_P(\pi^Y - y) \\ -D_B(\pi^W - w) \end{pmatrix}, \tag{4.7}$$

and the Hessian $\nabla^2 M$ may be written in the form

$$\begin{pmatrix} H + 2J(x)^{\mathrm{T}} D_P^{-1} J(x) & -2J(x)^{\mathrm{T}} D_P^{-1} & J(x)^{\mathrm{T}} & 0 \\ -2D_P^{-1} J(x) & 2(D_P^{-1} + D_B^{-1} W_B^{-1} \mathit{\Pi}^W + \mu^B S_B^{-1}) & -I_m & I_m \\ J(x) & -I_m & D_P & 0 \\ 0 & I_m & 0 & W_B^{-1}(D_B \mathit{\Pi}^W + \mu^B W_B^{-1} S_B) \end{pmatrix}, \tag{4.8}$$

where $H = H\big(x, \pi^Y + (\pi^Y - y)\big)$ and $\mathit{\Pi}^W = \operatorname{diag}(\pi^W)$.

At the start of iteration k, given the primal-dual iterate $v_k = (x_k,\, s_k,\, y_k,\, w_k)$, the search direction $\Delta v_k = (\Delta x_k,\, \Delta s_k,\, \Delta y_k,\, \Delta w_k)$ is computed by solving the linear equations

$$H_k^M \Delta v_k = -\nabla M(v_k), \tag{4.9}$$

where H_k^M is a positive-definite approximation of $\nabla^2 M(x_k,\; s_k,\; y_k,\; w_k)$. The remainder of this section focuses on the computation of the search direction for a single iteration, with the notation simplified by omitting the subscript k. The matrix H^M in the equations $H^M \Delta v = -\nabla M(v)$ is defined by substituting y for π^Y, w for π^W, s for s^E and a symmetric matrix $\widehat{H}$ for H in (4.8). This gives

$$H^M = \begin{pmatrix} \widehat{H} + 2J(x)^{\mathrm{T}} D_P^{-1} J(x) & -2J(x)^{\mathrm{T}} D_P^{-1} & J(x)^{\mathrm{T}} & 0 \\ -2D_P^{-1} J(x) & 2(D_P^{-1} + D_B^{-1}) & -I_m & I_m \\ J(x) & -I_m & D_P & 0 \\ 0 & I_m & 0 & D_B \end{pmatrix}, \tag{4.10}$$

where $\widehat{H}$ is chosen such that $\widehat{H} \approx H(x,y)$ and H^M is positive definite. A generalization of Theorem 5.1 of Gill, Kungurtsev and Robinson [46] may be used to show that the choice $\widehat{H} = H(x,y)$ is allowed in the neighborhood of a solution satisfying certain second-order optimality conditions. The approximate Newton equations (4.9) defined with H^M from (4.10) are not solved directly because of the potential for numerical instability. Instead, an *equivalent* transformed system is solved based on the transformation

$$UH^{M}\Delta v = -U\nabla M(v), \tag{4.11}$$

where U is a nonsingular matrix defined by

$$U = \begin{pmatrix} I_m & 0 & -2J(x)^{\mathrm{T}}D_{P}^{-1} & 0 \\ 0 & I_m & 2D_{P}^{-1} & -2D_{B}^{-1} \\ 0 & 0 & I_m & 0 \\ 0 & 0 & 0 & W + \mu^{B} I_m \end{pmatrix}. \tag{4.12}$$

Upon multiplication and application of the identity $W_{B}D_{B} = S_{B}$, the equations (4.11) may be rewritten as

$$\begin{pmatrix} \widehat{H} & 0 & -J(x)^{\mathrm{T}} & 0 \\ 0 & 0 & I_m & -I_m \\ J(x) & -I_m & D_{P} & 0 \\ 0 & W + \mu^{B} I_m & 0 & S + \mu^{B} I_m \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta s \\ \Delta y \\ \Delta w \end{pmatrix} = - \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}}y \\ y - w \\ c(x) - s + \mu^{P}(y - y^{E}) \\ s \cdot w - \mu^{B}(w^{E} - w + s^{E} - s) \end{pmatrix}. \tag{4.13}$$

These equations are identical to the shifted path-following equations (4.5) when $\widehat{H} = H(x,y)$. The solution of (4.13) is given by

$$\Delta w = y - w + \Delta y \quad \text{and} \quad \Delta s = -D_{B}(y + \Delta y) + \mu^{B} W_{B}^{-1}(w^{E} + s^{E} - s),$$

where Δx and Δy satisfy the equations

$$\begin{pmatrix} \widehat{H} & J(x)^{\mathrm{T}} \\ J(x) & -(D_{P} + D_{B}) \end{pmatrix} \begin{pmatrix} \Delta x \\ -\Delta y \end{pmatrix} = - \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}}y \\ D_{P}(y - \pi^{Y}) + D_{B}(y - \pi^{W}) \end{pmatrix}. \tag{4.14}$$

The matrix H^{M} in (4.10) is positive definite if $\widehat{H} + J(x)^{\mathrm{T}}(D_{P} + D_{B})^{-1}J(x)$ is positive definite or, equivalently, if the $(n+m)\times(n+m)$ matrix associated with (4.14) has inertia $(n, m, 0)$. If this condition does not hold for $\widehat{H} = H(x,y)$, a common choice of $\widehat{H}$ is the matrix $H(x,y) + \delta I_n$ for some positive scalar δ (see Section 5.3).

4.4 Minimizing the Merit Function using Projected Search

This section proposes a novel projected-search algorithm that utilizes a *non-monotone flexible quasi-Armijo* line search for minimizing the merit function $M(x,\ s,\ y,\ w\,;s^E,\ y^E,\ w^E,\ \mu^P,\ \mu^B)$ of (4.6) with fixed parameters s^E, y^E, w^E, μ^P and μ^B. The flexible quasi-Armijo line search is a generalization of the quasi-Armijo search that allows the acceptance of a step under a wider range of conditions. The generalization uses the idea of flexible line search proposed by Curtis and Nocedal [18], and also employs the relation between minimizing the merit function and finding a zero of the shifted path-following function $F(x,s,y,w\,;s^E,y^E,w^E,\mu^P,\mu^B)$ of (4.4). In this case, the notation is simplified by writing $M(x,s,y,w\,;s^E,y^E,w^E,\mu^P,\mu^B)$ and $F(x,s,y,w\,;s^E,y^E,w^E,\mu^P,\mu^B)$ as $M(v\,;\mu^P)$ and $F(v\,;\mu^P)$, respectively.

4.4.1 The algorithm

For the merit function $M(v\,;\mu^P)$ to be well-defined, the variables must satisfy the implicit bounds $s > -\mu^B e$, and $w > -\mu^B e$. Thus, minimizing the merit function $M(v\,;\mu^P)$ is equivalent to solving the bound-constrained problem

$$\underset{v}{\text{minimize}}\;\; M(v\,;\mu^P) \quad \text{subject to} \;\; v > \ell, \tag{IPBC'}$$

with $\ell = \big(-\infty, -\mu^B e, -\infty, -\mu^B e\big)$, where an entry of "$-\infty$" is used to indicate that the associated variable has no lower bound. Let $\mathbf{proj}_{\Omega_k}(v)$ be the projection of v onto the perturbed feasible region

$$\Omega_k = \big\{\, v : v \ge \min\big\{\, v_k - \sigma(v_k - \ell), 0 \,\big\}\,\big\}, \tag{4.15}$$

with σ a fixed positive scalar such that $0 < \sigma < 1$. The quantity σ may be interpreted as the "fraction to the boundary" parameter used in many conventional interior-point methods. The proposed projected-search method for problem (IPBC') is given in Algorithm 4. It generates a sequence of feasible iterates $\{v_k\}_{k=0}^{\infty}$ such that $v_{k+1} = \mathbf{proj}_{\Omega_k}(v_k + \alpha_k \Delta v_k)$, where Δv_k is the search direction computed as in Section 4.3, and α_k is a step computed using a flexible quasi-Armijo search.

To perform the flexible quasi-Armijo search, a line-search Armijo parameter μ^L is chosen such that $\mu^L \ge \mu^P$. At an iteration k, let $\psi_k(\alpha\,;\mu)$ and $\phi_k(\alpha\,;\mu)$ denote the functions $M\big(\mathbf{proj}_{\Omega_k}(v_k + \alpha\Delta v_k)\,;\mu\big)$ and $\big\|F\big(\mathbf{proj}_{\Omega_k}(v_k + \alpha\Delta v_k)\,;\mu\big)\big\|$. A step α_k is acceptable if all of the three conditions

$$\psi_k(\alpha_k\,;\mu^{\scriptscriptstyle P}) < \max\big\{\, \psi_k(0\,;\mu^{\scriptscriptstyle P}), M_{\max} \,\big\}, \tag{4.16a}$$

$$\psi_k(\alpha_k\,;\mu^{\scriptscriptstyle L}) < \max\big\{\, \psi_k(0\,;\mu^{\scriptscriptstyle L}), M_{\max} \,\big\}, \quad \text{and} \tag{4.16b}$$

$$\phi_k(\alpha_k\,;\mu^{\scriptscriptstyle P}) \le \eta_{\scriptscriptstyle F} \min\big\{\, \phi_k(0\,;\mu^{\scriptscriptstyle P}), \eta_{\scriptscriptstyle F}^{m_k} F_{\max} \,\big\} \tag{4.16c}$$

are satisfied, or

$$\psi_k(\alpha_k\,;\mu_k^{\scriptscriptstyle F}) \le \psi_k(0\,;\mu_k^{\scriptscriptstyle F}) + \alpha_k\eta_{\scriptscriptstyle A}\nabla M(v_k\,;\mu^{\scriptscriptstyle P})^{\mathrm{T}}\Delta v_k, \tag{4.16d}$$

for some value $\mu_k^{\scriptscriptstyle F} \in [\mu^{\scriptscriptstyle P}, \mu^{\scriptscriptstyle L}]$ and some positive $\eta_{\scriptscriptstyle F} < 1$. In these conditions, $M_{\max}$ and $F_{\max}$ are large preassigned parameters and m_k is the number of iterations prior to iteration k at which (4.16a)–(4.16c) were satisfied. Any α_k satisfying the conditions (4.16a)–(4.16c) or the condition (4.16d) is classified as a flexible quasi-Armijo step. Alternatively, an α_k that satisfies (4.16d) for $\mu_k^{\scriptscriptstyle F} = \mu^{\scriptscriptstyle P}$ is simply known as a quasi-Armijo step. The conditions (4.16a)–(4.16d) allow a step to be accepted if either (4.16a)–(4.16c) holds, which implies that α_k gives a sufficient decrease in the norm of the shifted path-following function F (4.4), or (4.16d) holds, which implies that α_k satisfies a flexible variant of the quasi-Armijo condition for the minimization of M.

The convergence analysis in subsection 4.4.2 below establishes the convergence of Algorithm 4 under typical assumptions. However, the ultimate purpose is to develop a practical algorithm for the solution of problem (NIPs) that uses Algorithm 4 as a basis for minimizing the underlying merit function. The slack-variable reset in Step 18 of Algorithm 4 plays a crucial role in this more general algorithm for handling (locally) infeasible problems (see Lemma 4.5.5). Analogous slack-variable resets are used in Gill, Murray and Saunders [50], and Gill, Kungurtsev and Robinson [46]. As defined in Step 17 of Algorithm 4, $\widehat{s}_{k+1}$ is the unique minimizer, with respect to s, of the sum of the terms (B), (C), (D), (G) and (H) in the definition of the function M. In particular, it follows from Step 17 and Step 18 of Algorithm 4 that the value of s_{k+1} computed in Step 18 satisfies

$$s_{k+1} \ge \widehat{s}_{k+1} = c(x_{k+1}) - \mu_k^{\scriptscriptstyle F}\big(y^{\scriptscriptstyle E} + \tfrac{1}{2}(w_{k+1} - y_{k+1}) + \mu^{\scriptscriptstyle B}\big),$$

which implies, after rearrangement, that

$$c(x_{k+1}) - s_{k+1} \le \mu_k^{\scriptscriptstyle F}\big(y^{\scriptscriptstyle E} + \tfrac{1}{2}(w_{k+1} - y_{k+1}) + \mu^{\scriptscriptstyle B}\big). \tag{4.17}$$

Algorithm 4 Minimizing M for fixed parameters s^E, y^E, w^E, μ^P, μ^B and μ^L.

```
procedure MERIT-PROJ$(x_0, s_0, y_0, w_0, s^E, w^E, \mu^P, \mu^B, \mu^L)$
    Restrictions: $s_0 + \mu^B e > 0$, $w_0 + \mu^B e > 0$, $s^E + w^E + \mu^B e > 0$, $\mu^L \ge \mu^P > 0$, $\mu^B > 0$;
    Constants: $\{ \eta_A, \gamma_A, \eta_F \} \in (0, 1)$;
    Set $v_0 \leftarrow (x_0, s_0, y_0, w_0)$;
    while $\|\nabla M(v_k)\| > 0$ do
        Choose $H_k^M \succ 0$, and then compute the search direction $\Delta v_k$ from (4.9);
        Set $\alpha_k \leftarrow 1$;
        loop
            if (4.16a)–(4.16c) hold or (4.16d) holds for $\mu_k^F = \mu^L$ then
                break;
            else if (4.16d) holds for $\mu_k^F = \mu^P$ then
                break;
            end if
            Set $\alpha_k \leftarrow \gamma_A \alpha_k$;
        end loop
        Set $v_{k+1} \leftarrow \mathbf{proj}_{\Omega_k}(v_k + \alpha_k \Delta v_k)$;
        Set $\widehat{s}_{k+1} \leftarrow c(x_{k+1}) - \mu_k^F\big(y^E + \frac{1}{2}(w_{k+1} - y_{k+1}) + \mu^B\big)$;
        Perform a slack reset $s_{k+1} \leftarrow \max\{s_{k+1}, \widehat{s}_{k+1}\}$;
        Set $v_{k+1} \leftarrow (x_{k+1}, s_{k+1}, y_{k+1}, w_{k+1})$;
    end while
end procedure
```

4.4.2 Convergence analysis

The following assumptions are made for the convergence analysis:

Assumption 4.4.1. *The functions f and c are twice continuously differentiable.*

Assumption 4.4.2. *The sequence of matrices $\{H_k^M\}_{k\ge 0}$ used in* (4.9) *are chosen to be uniformly positive definite and bounded in norm.*

Assumption 4.4.3. *The sequence of iterates $\{x_k\}$ is contained in a bounded set.*

Additionally, it will be show in Section 4.5 (proof of Lemma 4.5.2) that μ_k^F is fixed for all k sufficiently large if μ^L is chosen appropriately. In this section, without loss of generality, the parameter μ_k^F in Algorithm 4 is assumed to be fixed at a value μ^F, with either $\mu^F = \mu^P$ or $\mu^F = \mu^L$. In order to simplify the notation, let $M(v\,; \mu^F)$ denote $M(x,\, s,\, y,\, w\,; s^E,\, y^E,\, w^E,\, z^E,\, \mu^F,\, \mu^B)$.

Lemma 4.4.1. *The sequence of iterates $\{v_k\}$ computed by* Algorithm 4 *is such that $\big\{M(v_k\,; \mu^F)\big\}$ is bounded. In particular, if α_k is a step that satisfies* (4.16d)*, then $M(v_{k+1}\,; \mu^F) < M(v_k\,; \mu^F)$.*

Proof. As H_k^M is positive definite by Assumption 4.4.2 and $\nabla M(v_k\,; \mu^P)$ is assumed to be nonzero for all $k \ge 0$, the vector Δv_k is a descent direction for M at v_k. This property, together with equations

(4.16a) and (4.16b), imply that the line search performed in Algorithm 4 produces an α_k such that the new point $v_{k+1} = \mathbf{proj}_{\Omega_k}(v_k + \alpha_k \Delta v_k)$ satisfies $M(v_{k+1}\,; \mu^F) < \max\{M(v_k\,; \mu^F), M_{\max}\}$. In particular, if (4.16d) holds, then $M(v_{k+1}\,; \mu^F) < M(v_k\,; \mu^F)$. It follows that the only way that the desired result cannot hold is if the slack-reset procedure of Step 18 of Algorithm 4 causes M to increase. The proof is complete if it can be shown that this cannot happen.

The vector $\widehat{s}_{k+1}$ used in the slack reset is the unique minimizer of the sum of the terms (B), (C), (D), (G) and (H) defining the function $M(v\,; \mu^F)$, so that the sum of these terms cannot increase. Also, (A) is independent of s, so that its value does not change. The slack-reset procedure has the effect of possibly increasing the value of some of its components, which means that the (E) and (F) terms in the definition of M can only decrease. In total, this implies that the slack reset can never increase the value of M, which completes the proof. □

Lemma 4.4.2. *The sequence of iterates* $\{v_k\} = \{(x_k, s_k, y_k, w_k)\}$ *computed by* Algorithm 4 *satisfies the following properties.*

(i) *The sequences* $\{s_k\}$, $\{c(x_k) - s_k\}$, $\{y_k\}$, *and* $\{w_k\}$ *are bounded.*

(ii) *For every i it holds that*

$$\liminf_{k\geq 0}\, [\, s_k + \mu^B e\,]_i > 0 \;\; \textit{and} \;\; \liminf_{k\geq 0}\, [\, w_k + \mu^B e\,]_i > 0.$$

(iii) *The sequences* $\big\{\pi^Y(x_k, s_k)\big\}$, $\big\{\pi^W(s_k)\big\}$, *and* $\big\{\nabla M(v_k\,; \mu^P)\big\}$ *are bounded.*

(iv) *There exists a scalar M_{low} such that $M(v_k\,; \mu^F) \geq M_{\text{low}} > -\infty$ for all k.*

Proof. First, we consider the case where (4.16c) holds only finitely many times. For a proof by contradiction, assume that $\{s_k\}$ is unbounded. As $s_k + \mu^B e > 0$ by construction, there exists a subsequence of iterations $\mathcal{S}$ and component i such that

$$\lim_{k\in\mathcal{S}}\, [\, s_k\,]_i = \infty \;\; \text{and} \;\; [\, s_k\,]_i \geq [\, s_k\,]_j \;\; \text{for every } j \text{ and all } k\in\mathcal{S}. \tag{4.18}$$

Next it will be shown that M must go to infinity on $\mathcal{S}$. It follows from (4.18), Assumption 4.4.3, and the continuity of c that the term (A) in the definition of M is bounded below for all k, that (B) cannot go to $-\infty$ any faster than $\|s_k\|$ on $\mathcal{S}$, and that (C) converges to ∞ on $\mathcal{S}$ at the same rate as $\|s_k\|^2$. It is also clear that (D) is bounded below by zero. On the other hand, (E) goes to $-\infty$ on $\mathcal{S}$ at the rate $-\ln\big([\, s_k\,]_i + \mu^B\big)$. Next, note that (H) is bounded below. Now, if (F) is bounded below on $\mathcal{S}$, then the previous argument proves that M converges to infinity on $\mathcal{S}$, which contradicts Lemma 4.4.1. Otherwise, if (F) goes to $-\infty$ on $\mathcal{S}$, then (G) converges to ∞ faster than (F) converges to $-\infty$. Thus, M converges to ∞ on $\mathcal{S}$, which contradicts Lemma 4.4.1. We have thus proved that $\{s_k\}$ is bounded, which is the first part of result (i). The second part of (i), i.e., the uniform boundedness of $\{c(x_k) - s_k\}$, follows from the first result, the continuity of c, and Assumption 4.4.3.

The next step is to establish the third bound in part (i), i.e., that $\{y_k\}$ is bounded. For a proof by contradiction, assume that there exists some subsequence $\mathcal{S}$ and component i such that

$$\lim_{k\in\mathcal{S}} \bigl|[\, y_k \,]_i\bigr| = \infty \quad \text{and} \quad \bigl|[\, y_k \,]_i\bigr| \geq \bigl|[\, y_k \,]_j\bigr| \ \text{ for every } j \text{ and all } k \in \mathcal{S}.$$

Using arguments similar to those of the preceding paragraph, together with the result established above that $\{s_k\}$ is bounded, it follows that (A), (B) and (C) are bounded below over all k, and that (D) converges to ∞ on $\mathcal{S}$ at the rate of $[\, y_k \,]_i^2$ because $\{s_k\}$ is bounded, as has been shown above. Using the uniform boundedness of $\{\, s_k \,\}$ and the assumption that $s^E + w^E + \mu^B > 0$, it may be deduced that (E) is bounded below. If (F) is bounded below on $\mathcal{S}$, then (G) is also bounded, and as (H) is bounded below by zero we would conclude, in totality, that $\lim_{k\in\mathcal{S}} M(v_k) = \infty$, which contradicts Lemma 4.4.1. Thus, (F) must converge to $-\infty$, which implies that (G) converges to ∞ faster than (F) converges to $-\infty$, so that $\lim_{k\in\mathcal{S}} M(v_k\,;\mu^F) = \infty$ on $\mathcal{S}$, which contradicts Lemma 4.4.1. Thus, $\{\, y_k \,\}$ is bounded.

We now establish the final bound in part (i), i.e., we show that $\{\, w_k \,\}$ is bounded. The boundedness of $\{\, x_k \,\}$, $\{\, s_k \,\}$ and $\{\, y_k \,\}$ imply that (A), (B), (C), (D) and (H) are bounded and that (E) is bounded below. For a proof by contradiction, assume that the set is unbounded, which implies the existence of a subsequence $\mathcal{S}$ and a component i such that

$$\lim_{k\in\mathcal{S}} [\, w_k \,]_i = \infty.$$

Then (F) converges to $-\infty$, while (G) converges to ∞ faster than (F) converges to $-\infty$, so that $\lim_{k\in\mathcal{S}_1} M(v_k\,;\mu^F) = \infty$ on $\mathcal{S}$, which contradicts Lemma 4.4.1. It follows that $\{\, w_k \,\}$ is bounded.

Part (ii) is also proved by contradiction. Suppose that $\{\, [\, s_k + \mu^B e \,]_i \,\} \to 0$ on some subsequence $\mathcal{S}$ and for some component i. As before, (A), (B), (C), (D), (G) and (H) are all bounded from below over all k. We may also use $w^E + s^E + \mu^B > 0$ and the fact that $\{\, s_k \,\}$ and $\{\, w_k \,\}$ were proved to be bounded in part (i) to conclude that (E) and (F) converge to ∞ on $\mathcal{S}$. It follows that $\lim_{k\in\mathcal{S}} M(v_k\,;\mu^F) = \infty$, which contradicts Lemma 4.4.1, and therefore establishes that $\liminf\, [\, s_k + \mu^B e \,]_i > 0$ for every $1 \leq i \leq m$. A similar argument may be used to prove that

$\liminf\, [\, w_k + \mu^B e\,]_i > 0$ for every $1 \le i \le m$, which completes the proof.

Part (iii) and Part (iv) can be proved similarly as in the proof of Lemma 3.2(iii) and (iv) in [46]. Consider part (iii). The sequence $\big\{\pi^Y(x_k, s_k)\big\}$ is bounded as a consequence of part (i) and the fact that y^E and μ^P are fixed. Similarly, the sequence $\big\{\pi^W(s_k)\big\}$ is bounded as a consequence of part (ii) and the fact that w^E, s^E and μ^B are fixed. Lastly, the sequence $\big\{\nabla M(x_k, s_k, y_k)\big\}$ is bounded as a consequence of parts (i) and (ii), the uniform boundedness just established for $\big\{\pi^Y(x_k, s_k)\big\}$ and $\big\{\pi^W(s_k)\big\}$, Assumption 4.4.1, Assumption 4.4.3, and the fact that y^E, w^E, s^E, μ^P, and μ^B are fixed. For part (iv) it will be shown that each term in the definition of M is bounded below. Term (A) is bounded below because of Assumption 4.4.1 and Assumption 4.4.2. Term (B) is bounded below as a consequence of part (i) and the fact that y^E is kept fixed. Terms (C) and (D) are both nonnegative, hence, trivially bounded below. Terms (E) and (F) are bounded below because μ^B and $w^E + s^E + \mu^B e > 0$ are held fixed, and part (i). Term (G) is bounded below because of part (i). Finally, (H) is bounded below because $s > -\mu^B e$. The existence of the lower bound M_{low} now follows. □

Certain results hold when the gradient of $M(v\,; \mu^P)$ is bounded away from zero.

Lemma 4.4.3. *If there exists a positive scalar ϵ and a subsequence of iterates $\mathcal{S}$ satisfying*

$$\|\nabla M(v_k\,; \mu^P)\| \ge \epsilon \;\textit{ for all }\; k \in \mathcal{S},$$

then the following results must hold.

(i) *The set $\big\{\, \|\varDelta v_k\|\, \big\}_{k\in\mathcal{S}}$ is bounded above and bounded away from zero.*

(ii) *There exists a positive scalar δ such that $\nabla M(v_k\,; \mu^P)^T \varDelta v_k \le -\delta$ for all $k \in \mathcal{S}$.*

Proof. See the proof of Lemma 3.3 in [46]. □

Now the main convergence result for Algorithm 4 can be established.

Theorem 4.4.1 (Flexible quasi-Armijo search)**.** *Under* Assumptions 4.4.1–4.4.3*, there exists an iteration subsequence $\mathcal{S}$ such that*

$$\lim_{k\in\mathcal{S}} \nabla M(v_k\,; \mu^P) = 0.$$

Proof. First, consider the case where there exists an infinite subsequence of iterates $\mathcal{S}$ such that the line-search conditions (4.16a)–(4.16c) hold for all $k \in \mathcal{S}$. Then the line-search condition (4.16c) implies that $\lim_{k\in\mathcal{S}} \|F(v_k\,; \mu^P)\| = 0$. By (4.11), $F(v_k\,; \mu^P) = U_k \nabla M(v_k\,; \mu^P)$, where U_k is a matrix of the form (4.12). Lemma 4.4.2((ii)) implies that $\{\, \|U_k\| \,\}$ is uniformly bounded away from zero, which ensures that $\lim_{k\in\mathcal{S}} \nabla M(v_k\,;\mu^P) = 0$.

Now assume the complementary case where the subsequence of iterates such that the line-search conditions (4.16a)–(4.16c) hold is finite. This implies that there exists k_0 such that for all $k > k_0$, the line-search condition (4.16d) must hold. Thus, all the subsequent iterates $\{\, v_k \,\}_{k>k_0}$ lies within the level set

$$\mathcal{L}\big(M(v_{k_0}\,;\mu^F)\big) \triangleq \{\, v \in \Omega : M(v\,;\mu^F) \le M(v_{k_0}\,;\mu^F) \,\},$$

where Ω represents the open set in which the merit function $M(v\,;\mu^F)$ is well defined, i.e.,

$$\Omega = \{\, v = (x,s,y,w) : v > \ell \,\}, \text{ with } \ell = \big(-\infty, -\mu^B e, -\infty, -\mu^B e\big).$$

Notice that the value of $M(v\,;\mu^F)$ is $+\infty$ on the boundary of Ω. Then by the continuity of the function $M(v\,;\mu^F)$, the level set $\mathcal{L}\big(M(v_{k_0}\,;\mu^F)\big)$ is a closed subset of Ω. Moreover, Assumption 4.4.3 and Lemma 4.4.2(i) imply that the set of iterates $\{\, v_k \,\}_{k>k_0}$ is a bounded subset of $\mathcal{L}\big(M(v_{k_0}\,;\mu^F)\big)$. Hence, there exists a compact subset of Ω such that $\{\, v_k \,\}_{k>k_0}$ lies within the compact subset. It follows that

$$\kappa \triangleq \min_{k>k_0, 1\le i\le n} \{\, [\, v_k\,]_i - [\,\ell\,]_i \,\} > 0.$$

We show by contradiction that $\lim_{k\to\infty} \nabla M(v_k\,;\mu^P) = 0$. Suppose there exists a constant $\epsilon > 0$ and a subsequence $\mathcal{G}$ such that $\|\nabla M(v_k\,;\mu^P)\| \ge \epsilon$ for all $k \in \mathcal{G}$. It follows from Lemma 4.4.1 and Lemma 4.4.2(iv) that $\lim_{k\to\infty} M(v_k\,;\mu^F) = M_{\min} > -\infty$. Using this result and the assumption that the line-search condition (4.16d) is satisfied for all k sufficiently large, it must follow that

$$\lim_{k\to\infty} \alpha_k \nabla M(v_k\,;\mu^P)^{\mathrm{T}} \Delta v_k = 0,$$

which, together with Lemma 4.4.3(ii), implies that $\lim_{k\in\mathcal{G}} \alpha_k = 0$. For each k, define $\beta_k \triangleq \alpha_k/\gamma_A$. Then $\lim_{k\in\mathcal{G}} \beta_k = 0$ and the backtracking procedure in Algorithm 4 implies that the condition (4.16d) does not hold for the step β_k for all k sufficiently large. This means that the more stringent quasi-Armijo condition does not hold, i.e.,

$$M\big(\mathbf{proj}_{\,\Omega_k}(v_k + \beta_k \Delta v_k)\,;\mu^P\big) > M\big(v_k\,;\mu^P\big) + \alpha_k \eta_A \nabla M(v_k\,;\mu^P)^{\mathrm{T}} \Delta v_k \tag{4.19}$$

for all k sufficiently large. By Lemma 4.4.3(i), we also have $\lim_{k\in\mathcal{G}} \|\beta_k \Delta v_k\| = 0$. Thus, there exists $\bar{k}$ such that every component of $\beta_k \Delta v_k$ satisfies $|[\,\beta_k \Delta v_k\,]_i| < \sigma\gamma$ for all $k > \bar{k}$ in $\mathcal{G}$. It follows that

$v_k + \beta_k \mathit{\Delta} v_k \in \mathit{\Omega}_k$, which implies $\mathbf{proj}_{\,\mathit{\Omega}_k}(v_k + \beta_k \mathit{\Delta} v_k) = v_k + \beta_k \mathit{\Delta} v_k$. Now let $\mathcal{G}'$ denote the indices $k > \max\{\, k_0, \bar{k} \,\}$ of iterations at which a reduction in the initial step length was necessary, i.e., $\mathcal{G}' = \{\, k : \alpha_k < 1,\ k \in \mathcal{G},\ k > \max\{\, k_0, \bar{k} \,\} \,\}$. As α_k converges to zero, $\mathcal{G}'$ must be an infinite set. The inequality (4.19) implies that

$$M(v_k + \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) > M(v_k\,;\mu^{\scriptscriptstyle P}) + \beta_k \eta_{\scriptscriptstyle A} \nabla M(v_k\,;\mu^{\scriptscriptstyle P})^{\mathrm{T}} \mathit{\Delta} v_k$$

for all k in $\mathcal{G}'$. Adding $-\beta_k \nabla M(v_k\,;\mu^{\scriptscriptstyle P})^{\mathrm{T}} \mathit{\Delta} v_k$ to both sides and rearranging gives

$$\begin{aligned} M(v_k + \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - M(v_k\,;\mu^{\scriptscriptstyle P}) - \beta_k \nabla M(v_k\,;\mu^{\scriptscriptstyle P})^{\mathrm{T}} \mathit{\Delta} v_k &> -\beta_k(1-\eta_{\scriptscriptstyle A}) \nabla M(v_k\,;\mu^{\scriptscriptstyle P})^{\mathrm{T}} \mathit{\Delta} v_k \\ &> \beta_k(1-\eta_{\scriptscriptstyle A})\delta, \ \text{ for all } \ k \in \mathcal{G}'. \end{aligned} \tag{4.20}$$

The Taylor expansion of $M(v_k + \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P})$ gives

$$\begin{aligned} &M(v_k + \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - M(v_k\,;\mu^{\scriptscriptstyle P}) - \beta_k \nabla M(v_k\,;\mu^{\scriptscriptstyle P})^{\mathrm{T}} \mathit{\Delta} v_k \\ &\qquad = \beta_k \int_0^1 \big(\nabla M(v_k + \tau \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \big)^{\mathrm{T}} \mathit{\Delta} v_k \, d\tau. \end{aligned} \tag{4.21}$$

If $\|\cdot\|_{\scriptscriptstyle D}$ denotes the norm dual to $\|\cdot\|$, i.e., $\|v\|_{\scriptscriptstyle D} = \max_{u \ne 0} |v^{\mathrm{T}} u|/\|u\|$, then

$$\begin{aligned} &\big| \big(\nabla M(v_k + \tau \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \big)^{\mathrm{T}} \mathit{\Delta} v_k \big| \\ &\qquad \le \| \nabla M(v_k + \tau \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \|_{\scriptscriptstyle D} \|\mathit{\Delta} v_k\|. \end{aligned}$$

If this inequality is substituted in (4.21), it then follows from (4.20) that

$$\begin{aligned} (1-\eta_{\scriptscriptstyle A})\delta &< \int_0^1 \big(\nabla M(v_k + \tau \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \big)^{\mathrm{T}} \mathit{\Delta} v_k \, d\tau \\ &\le \max_{0 \le \tau \le 1} \| \nabla M(v_k + \tau \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \|_{\scriptscriptstyle D} \|\mathit{\Delta} v_k\|, \ \text{ for all } \ k \in \mathcal{G}'. \end{aligned}$$

The continuity of ∇M implies that there exists some $\tau_k \in [0, \beta_k]$ such that

$$\max_{0 \le \tau \le 1} \| \nabla M(v_k + \tau \beta_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \|_{\scriptscriptstyle D} = \| \nabla M(v_k + \tau_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \|_{\scriptscriptstyle D}.$$

Then

$$(1-\eta_{\scriptscriptstyle A})\delta < \| \nabla M(v_k + \tau_k \mathit{\Delta} v_k\,;\mu^{\scriptscriptstyle P}) - \nabla M(v_k\,;\mu^{\scriptscriptstyle P}) \|_{\scriptscriptstyle D} \|\mathit{\Delta} v_k\|. \tag{4.22}$$

However, $\alpha_k \Delta v_k \to 0$ implies $\tau_k \Delta v_k \to 0$ for $k \in \mathcal{G}$, and the continuity of ∇M gives

$$\|\nabla M(v_k + \tau_k \Delta v_k \,; \mu^P) - \nabla M(v_k \,; \mu^P)\|_D \to 0.$$

Lemma 4.4.3(i) implies that the right-hand side of (4.22) converges to zero, which gives the required contradiction. □

4.5 Solving the Nonlinear Optimization Problem

In this section, a projected-search interior method for solving the nonlinear optimization problem (NIPs) is formulated and analyzed. The method incorporates the projected-search algorithm presented in Section 4.4 with strategies for adjusting the parameters in the definition of the merit function. These parameters were fixed in Algorithm 4.

4.5.1 The algorithm

The proposed method is given in Algorithm 5. The method uses the distinction among O-iterations, M-iterations and F-iterations, which are described below.

The definition of an O-iteration is based on the optimality conditions for problem (NIPs). Progress towards optimality of the iterate $v_{k+1} = (x_{k+1},\, s_{k+1},\, y_{k+1},\, w_{k+1})$ is defined in terms of the following feasibility, stationarity, and complementarity measures:

$$\begin{aligned}
\chi_{\text{feas}}(v_{k+1}) &= \|c(x_{k+1}) - s_{k+1}\|, \\
\chi_{\text{stny}}(v_{k+1}) &= \max\bigl(\|\nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}} y_{k+1}\|, \|y_{k+1} - w_{k+1}\|\bigr), \ \text{ and} \\
\chi_{\text{comp}}(v_{k+1}, \mu_k^B) &= \bigl\|\min\bigl(q_1(v_{k+1}), q_2(v_{k+1}, \mu_k^B)\bigr)\bigr\|,
\end{aligned}$$

where

$$\begin{aligned}
q_1(v_{k+1}) &= \max\bigl(|\min(s_{k+1}, w_{k+1}, 0)|, |s_{k+1} \cdot w_{k+1}|\bigr), \ \text{ and} \\
q_2(v_{k+1}, \mu_k^B) &= \max\bigl(\mu_k^B e, |\min(s_{k+1} + \mu_k^B e, w_{k+1} + \mu_k^B e, 0)|, |(s_{k+1} + \mu_k^B e) \cdot (w_{k+1} + \mu_k^B e)|\bigr).
\end{aligned}$$

A first-order KKT point v_{k+1} for problem (NIPs) satisfies $\chi(v_{k+1}, \mu_k^B) = 0$, where

$$\chi(v, \mu) = \chi_{\text{feas}}(v) + \chi_{\text{stny}}(v) + \chi_{\text{comp}}(v, \mu). \tag{4.23}$$

Given these definitions, the kth iteration is designated as an O-iteration if $\chi(v_{k+1}, \mu_k^B) \le \chi_k^{\max}$, where $\{\,\chi_k^{\max}\,\}$ is a monotonically decreasing positive sequence. At an O-iteration the parameters are updated as $y_{k+1}^E = y_{k+1}$, $w_{k+1}^E = w_{k+1}$ and $\chi_{k+1}^{\max} = \frac{1}{2}\chi_k^{\max}$ (see Step 11 of Algorithm 5). These updates ensure that $\{\,\chi_k^{\max}\}$ converges to zero if infinitely many O-iterations occur. The point v_{k+1} is called an O-iterate.

If the condition for an O-iteration does not hold, a test is made to determine if $v_{k+1} = (x_{k+1}, s_{k+1}, y_{k+1}, w_{k+1})$ is an approximate first-order solution of the problem

$$\underset{v=(x,s,y,w)}{\text{minimize}} \;\; M(v\,; s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B). \tag{4.24}$$

In particular, the kth iteration is called an M-iteration if v_{k+1} satisfies

$$\|\nabla_x M(v_{k+1}\,; s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B)\|_\infty \le \tau_k, \tag{4.25a}$$

$$\|\nabla_s M(v_{k+1}\,; s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B)\|_\infty \le \tau_k, \tag{4.25b}$$

$$\|\nabla_y M(v_{k+1}\,; s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B)\|_\infty \le \tau_k \|D_{k+1}^P\|_\infty, \;\; \text{and} \tag{4.25c}$$

$$\|\nabla_w M(v_{k+1}\,; s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B)\|_\infty \le \tau_k \|D_{k+1}^B\|_\infty, \tag{4.25d}$$

where τ_k is a positive tolerance, $D_{k+1}^P = \mu_k^P I$, and $D_{k+1}^B = (S_{k+1} + \mu_k^B I)(W_{k+1} + \mu_k^B I)^{-1}$. In this case v_{k+1} is called an M-iterate because it is an approximate first-order solution of (4.24). The estimates s_{k+1}^E, y_{k+1}^E and w_{k+1}^E are defined by the safeguarded values

$$\left.\begin{aligned} s_{k+1}^E &= \min\big(\max(0, s_{k+1}), s_{\max} e\big), \\ y_{k+1}^E &= \max\big(-y_{\max} e, \min(y_{k+1}, y_{\max} e)\big), \\ w_{k+1}^E &= \min\big(w_{k+1}, w_{\max} e\big) \end{aligned}\right\} \tag{4.26}$$

for some large positive constants $s_{\max}$, $y_{\max}$ and $w_{\max}$. Next, Step 15 checks if the condition

$$\chi_{\text{feas}}(v_{k+1}) \le \tau_k \tag{4.27}$$

holds. If the condition holds, then $\mu_{k+1}^P \leftarrow \mu_k^P$; otherwise, $\mu_{k+1}^P \leftarrow \frac{1}{2}\mu_k^P$ to place more emphasis on satisfying the constraint $c(x) - s = 0$ in subsequent iterations. Similarly, Step 16 checks the inequalities

$$\chi_{\text{comp}}(v_{k+1}, \mu_k^B) \le \tau_k, \quad s_{k+1} \ge -\tau_k e, \quad \text{and} \quad w_{k+1} \ge -\tau_k e. \tag{4.28}$$

If these conditions hold, then $\mu_{k+1}^B \leftarrow \mu_k^B$; otherwise, $\mu_{k+1}^B \leftarrow \frac{1}{2}\mu_k^B$ to place more emphasis on achieving complementarity in subsequent iterations.

An iteration that is not an O- or M-iteration is called an F-iteration. In an F-iteration none of the parameters in the merit function are changed, so that progress is measured solely in terms of the reduction in the merit function.

Algorithm 5 An all-shifted projected-search interior method.

```
 1: procedure pdProj($x_0, s_0, y_0, w_0$)
 2:     Restrictions: $s_0 > 0$ and $w_0 > 0$;
 3:     Constants: $\{\, \eta_A, \gamma_A \,\} \subset (0,1)$ and $\{\, y_{\max}, w_{\max}, s_{\max} \,\} \subset (0,\infty)$;
 4:     Choose $y_0^E$;   $\chi_0^{\max} > 0$; $\{\, \mu_0^P, \mu_0^B \,\} \subset (0,\infty)$; and $\mu_0^L \ge \mu_0^P$;
 5:     Choose $w_0^E$ and $s_0^E$ such that $w_0^E + s_0^E + \mu_0^B e > 0$;
 6:     Set $v_0 = (x_0, s_0, y_0, w_0)$;   $k \leftarrow 0$;
 7:     while $\|\nabla M(v_k)\| > 0$ do
 8:         $(s^E, y^E, w^E, \mu^P, \mu^B) \leftarrow (s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B)$;
 9:         Compute $v_{k+1}$ in Steps 6–19 of Algorithm 4;
10:         if $\chi(v_{k+1}, \mu_k^B) \le \chi_k^{\max}$ then                                    [O-iterate]
11:             $(\chi_{k+1}^{\max}, y_{k+1}^E, w_{k+1}^E, \mu_{k+1}^P, \mu_{k+1}^B, \tau_{k+1}) \leftarrow (\frac{1}{2}\chi_k^{\max}, y_{k+1}, w_{k+1}, \mu_k^P, \mu_k^B, \tau_k)$;
12:             $s_{k+1}^E \leftarrow \max\{\, 0, s_{k+1} \,\}$;
13:         else if $v_{k+1}$ satisfies (4.25a)–(4.25d) then                                  [M-iterate]
14:             Set $(\chi_{k+1}^{\max}, \tau_{k+1}) = (\chi_k^{\max}, \frac{1}{2}\tau_k)$;   Set $s_{k+1}^E$, $y_{k+1}^E$ and $w_{k+1}^E$ using (4.26);
15:             if $\chi_{\rm feas}(v_{k+1}) \le \tau_k$ then $\mu_{k+1}^P \leftarrow \mu_k^P$ else $\mu_{k+1}^P \leftarrow \frac{1}{2}\mu_k^P$ end if
16:             if $\chi_{\rm comp}(v_{k+1}, \mu_k^B) \le \tau_k$,  $s_{k+1} \ge -\tau_k e$  and  $w_{k+1} \ge -\tau_k e$ then
17:                 $\mu_{k+1}^B \leftarrow \mu_k^B$;
18:             else
19:                 $\mu_{k+1}^B \leftarrow \frac{1}{2}\mu_k^B$;
20:                 Reset $s_{k+1}$ and $w_{k+1}$ so that $s_{k+1} + \mu_{k+1}^B e > 0$ and $w_{k+1} + \mu_{k+1}^B e > 0$;
21:             end if
22:         else                                                                               [F-iterate]
23:             $(\chi_{k+1}^{\max}, s_{k+1}^E, y_{k+1}^E, w_{k+1}^E, \mu_{k+1}^P, \mu_{k+1}^B, \tau_{k+1}) \leftarrow (\chi_k^{\max}, s_k^E, y_k^E, w_k^E, \mu_k^P, \mu_k^B, \tau_k)$;
24:         end if
25:         Update $\mu_{k+1}^L$ as in (4.29);
26:     end while
27: end procedure
```

Reducing the barrier parameter μ^B in Step 19 of Algorithm 5 may cause a slack variable s_i or a dual variable w_i to become infeasible with respect to its shifted bounds. In Step 20, if a multiplier w_i becomes infeasible after μ^B is reduced, it is reinitialized as $\max\{\, y_i, \frac{1}{2}w_i \,\}$. To remedy the infeasibility of a slack variable s_i, suppose μ^B and $\bar{\mu}^B$ denote a shift before and after it is reduced, with $s_i + \mu^B > 0$ and $s_i + \bar{\mu}^B \le 0$, a strategy is proposed in Section 5.4 of [46], which temporarily imposes an equality constraint $s_i = 0$. This constraint is enforced by the primal-dual augmented Lagrangian term until the nonlinear constraint value $c_i(x)$ becomes larger than $\bar{\mu}^B$, at which point

s_i is assigned the value $s_i = c_i(x)$ and allowed to move. On being freed, the corresponding Lagrange multiplier w_i is reinitialized as $\max\{\, y_i, \epsilon \,\}$, where ϵ is a small positive constant.

Given an initial value $\mu_0^L \geq \mu_0^P$, in Step 25 of Algorithm 5, the line-search parameter μ_k^L is updated as

$$\mu_{k+1}^L = \begin{cases} \mu_k^L & \text{if } \psi_k(\alpha_k\,; \mu_k^L) \leq \psi_k(0\,; \mu_k^L) + \alpha_k \eta_A \delta_k \text{ and } \mu_{k+1}^P = \mu_k^P; \\ \max\{\, \frac{1}{2}\mu_k^L, \mu_{k+1}^P \,\} & \text{otherwise,} \end{cases} \tag{4.29}$$

where $\delta_k = \nabla M(v_k\,; \mu^P)^T \Delta v_k$. This updating rule guarantees that $\mu_k^L \geq \mu_k^P$ for all k.

4.5.2 Convergence Analysis

Convergence analysis for Algorithm 5 follows a similar procedure as in Section 4.2 of [46], which uses the properties of the *complementary approximate KKT (*CAKKT) condition proposed by Andreani, Martínez and Svaiter [2], as described below.

Definition 4.5.1 (CAKKT condition)**.** *A feasible point* (x^*, s^*) *(i.e., a point such that* $s^* \geq 0$ *and* $c(x^*) - s^* = 0$*) is said to satisfy the* CAKKT condition if there exists a sequence $\{\, (x_j, s_j, u_j, z_j) \,\}$ with $\{\, x_j \,\} \to x^*$ and $\{\, s_j \,\} \to s^*$ such that

$$\{\, \nabla f(x_j) - J(x_j)^T u_j \,\} \to 0, \tag{4.30}$$

$$\{\, u_j - z_j \,\} \to 0, \tag{4.31}$$

$$\{\, z_j \,\} \geq 0, \quad \text{and} \tag{4.32}$$

$$\{\, z_j \cdot s_j \,\} \to 0. \tag{4.33}$$

Any (x^*, s^*) satisfying these conditions is called a CAKKT point.

Theorem 4.5.1 (Andreani, Martínez, Ramos and Svaiter [1, Theorem 4.2])**.** *If* (x^*, s^*) *is a* CAKKT point that satisfies CAKKT-regularity, then (x^*, s^*) is a first-order KKT point for (NIPs).

The first part of the analysis concerns the conditions under which limit points of the sequence $\{\, (x_k, s_k) \,\}$ are CAKKT points. As the results are tied to the different iteration types, to facilitate referencing of the iterations during the analysis we define

$$\begin{aligned} \mathcal{O} &= \{\, k : \text{iteration } k \text{ is an O-iteration} \,\}, \\ \mathcal{M} &= \{\, k : \text{iteration } k \text{ is an M-iteration} \,\}, \quad \text{and} \\ \mathcal{F} &= \{\, k : \text{iteration } k \text{ is an F-iteration} \,\}. \end{aligned}$$

Lemma 4.5.1. *If $|\mathcal{O}| = \infty$ there exists at least one limit point (x^*, s^*) of the infinite sequence $\{\, (x_{k+1}, s_{k+1}) \,\}_{k\in\mathcal{O}}$ and any such limit point is a* CAKKT point.

Proof. Assumption 4.4.3 implies that there must exist at least one limit point of $\{\, x_{k+1} \,\}_{k\in\mathcal{O}}$. If x^* is such a limit point, Assumption 4.4.1 implies the existence of $\mathcal{K} \subseteq \mathcal{O}$ such that $\{\, x_{k+1} \,\}_{k\in\mathcal{K}} \to x^*$ and $\{\, c(x_{k+1}) \,\}_{k\in\mathcal{K}} \to c(x^*)$. As $|\mathcal{O}| = \infty$, the updating strategy of Algorithm 5 gives $\{\, \chi_k^{\max} \,\} \to 0$. Furthermore, as $\chi(v_{k+1}, \mu_k^B) \le \chi_k^{\max}$ for all $k \in \mathcal{K} \subseteq \mathcal{O}$, and $\chi_{\text{feas}}(v_{k+1}) \le \chi(v_{k+1}, \mu_k^B)$ for all k, it follows that $\{\, \chi_{\text{feas}}(v_{k+1}) \,\}_{k\in\mathcal{K}} \to 0$, i.e., $\{\, c(x_{k+1}) - s_{k+1} \,\}_{k\in\mathcal{K}} \to 0$. With the definition $s^* = c(x^*)$, it follows that $\{\, s_{k+1} \,\}_{k\in\mathcal{K}} \to \lim_{k\in\mathcal{K}} c(x_{k+1}) = c(x^*) = s^*$, which implies that (x^*, s^*) is feasible for the general constraints because $c(x^*) - s^* = 0$. The remaining feasibility condition $s^* \ge 0$ is proved componentwise. For any $1 \le i \le m$, define

$$\mathcal{Q}_1 = \{\, k : [q_1(v_{k+1})]_i \le [q_2(v_{k+1}, \mu_k^B)]_i \,\} \ \text{ and } \ \mathcal{Q}_2 = \{\, k : [q_2(v_{k+1}, \mu_k^B)]_i < [q_1(v_{k+1})]_i \,\},$$

where q_1 and q_2 are used in the definition of χ_{comp}. If the set $\mathcal{K} \cap \mathcal{Q}_1$ is infinite, then it follows from the inequalities $\{\, \chi_{\text{comp}}(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \le \{\, \chi(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \le \{\, \chi_k^{\max} \,\}_{k\in\mathcal{K}} \to 0$ that $s_i^* = \lim_{\mathcal{K}\cap\mathcal{Q}_1} \, [s_{k+1}]_i \ge 0$. Using a similar argument, if the set $\mathcal{K} \cap \mathcal{Q}_2$ is infinite, then $s_i^* = \lim_{\mathcal{K}\cap\mathcal{Q}_2} \, [s_{k+1}]_i = \lim_{\mathcal{K}\cap\mathcal{Q}_2} \, [s_{k+1} + \mu_k^B e]_i \ge 0$, where the second equality uses the limit $\{\, \mu_k^B \,\}_{k\in\mathcal{K}\cap\mathcal{Q}_2} \to 0$ that follows from the definition of $\mathcal{Q}_2$. Combining these two cases implies that $s_i^* \ge 0$, as claimed. It follows that the limit point (x^*, s^*) is feasible.

It remains to show that (x^*, s^*) is a CAKKT point. Let

$$[\bar{s}_{k+1}]_i = \begin{cases} [s_{k+1}]_i & \text{if } k \in \mathcal{Q}_1; \\ [s_{k+1} + \mu_k^B e]_i & \text{if } k \in \mathcal{Q}_2, \end{cases}$$

and

$$[\bar{w}_{k+1}]_i = \begin{cases} \max\{\, [w_{k+1}]_i, 0 \,\} & \text{if } k \in \mathcal{Q}_1; \\ [w_{k+1} + \mu_k^B e]_i & \text{if } k \in \mathcal{Q}_2, \end{cases}$$

for every $1 \leq i \leq m$, and consider the sequence $(x_{k+1},\, \bar{s}_{k+1},\, y_{k+1},\, \bar{w}_{k+1})_{k\in\mathcal{K}}$ as a candidate for the sequence used in Definition 4.5.1 to verify that (x^*, s^*) is a CAKKT point. If $\mathcal{O} \cap \mathcal{Q}_2$ is finite, then it follows from the definition of $\bar{s}_{k+1}$ and the limit $\{\, s_{k+1} \,\}_{k\in\mathcal{K}} \to s^*$ that $\{\, [\, \bar{s}_{k+1} \,]_i \,\}_{k\in\mathcal{K}} \to s_i^*$; also, $\{\, \chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \to 0$ implies that $\liminf_{k\in\mathcal{K}} [\, w_{k+1} \,]_i \geq 0$, therefore $\{\, [\, \bar{w}_{k+1} - w_{k+1} \,]_i \,\}_{k\in\mathcal{K}} \to 0$. On the other hand, if $\mathcal{O} \cap \mathcal{Q}_2$ is infinite, then the definitions of $\mathcal{Q}_2$ and $\chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B)$, together with the limit $\{\, \chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \to 0$ imply that $\{\, \mu_k^B \,\} \to 0$, giving $\{\, [\, \bar{s}_{k+1} \,]_i \,\}_{k\in\mathcal{K}} \to s_i^*$ and $\{\, [\, \bar{w}_{k+1} - w_{k+1} \,]_i \,\}_{k\in\mathcal{K}} \to 0$. As the choice of i was arbitrary, these cases taken together imply that $\{\, \bar{s}_{k+1} \,\}_{k\in\mathcal{K}} \to s^*$ and $\{\, \bar{w}_{k+1} - w_{k+1} \,\}_{k\in\mathcal{K}} \to 0$.

The next step is to show that $\{\, (x_{k+1}, \bar{s}_{k+1}, y_{k+1}, \bar{w}_{k+1}) \,\}_{k\in\mathcal{K}}$ satisfies the conditions required by Definition 4.5.1. It follows from the limit $\{\, \chi(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \to 0$ established above that $\{\, \chi_{\mathrm{stny}}(v_{k+1}) + \chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \leq \{\, \chi(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \to 0$. This, together with the limit $\{\, \bar{w}_{k+1} - w_{k+1} \,\}_{k\in\mathcal{K}} \to 0$, implies that $\{\, \nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}} y_{k+1} \,\}_{k\in\mathcal{K}} \to 0$ and $\{\, y_{k+1} - w_{k+1} \,\}_{k\in\mathcal{K}} \to 0$, which establishes that conditions (4.30) and (4.31) hold. The nonnegativity of $\bar{w}_{k+1}$ for all k is obvious from its definition, which implies that (4.32) is satisfied for $\{\, \bar{w}_k \,\}_{k\in\mathcal{K}}$. Finally, it must be shown that (4.33) holds, i.e., that $\{\, \bar{w}_{k+1} \cdot \bar{s}_{k+1} \,\}_{k\in\mathcal{K}} \to 0$. Consider the ith components of $\bar{s}_k$ and $\bar{w}_k$. If the set $\mathcal{K} \cap \mathcal{Q}_1$ is infinite, then the definitions of $\bar{s}_{k+1}$, $q_1(v_{k+1})$ and $\chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B)$, together with the limit $\{\, \chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \to 0$, imply that $\{\, [\, \bar{w}_{k+1} \cdot \bar{s}_{k+1} \,]_i \,\}_{\mathcal{K}\cap\mathcal{Q}_1} \to 0$. Similarly, if the set $\mathcal{K} \cap \mathcal{Q}_2$ is infinite, then the definitions of $\bar{s}_{k+1}$, $q_2(v_{k+1}, \mu_k^B)$ and $\chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B)$, together with the limits $\{\, \chi_{\mathrm{comp}}(v_{k+1}, \mu_k^B) \,\}_{k\in\mathcal{K}} \to 0$ and $\{\, \bar{w}_{k+1} - w_{k+1} \,\}_{k\in\mathcal{K}} \to 0$, imply that $\{\, [\, \bar{w}_{k+1} \cdot \bar{s}_{k+1} \,]_i \,\}_{k\in\mathcal{K}\cap\mathcal{Q}_2} \to 0$. Thus, these two cases lead to the conclusion that $\{\, \bar{w}_{k+1} \cdot \bar{s}_{k+1} \,\}_{k\in\mathcal{K}} \to 0$, which implies that condition (4.33) is satisfied. This concludes the proof that (x^*, s^*) is a CAKKT point. □

In the complementary case where $|\mathcal{O}| < \infty$, it will be shown that every limit point of the iteration subsequence $\{\, (x_{k+1}, s_{k+1}) \,\}_{k\in\mathcal{M}}$ is infeasible with respect to the constraint $c(x) - s = 0$ but solves the least-infeasibility problem

$$\underset{x,s}{\text{minimize}} \;\; \tfrac{1}{2}\|c(x) - s\|_2^2 \;\; \text{subject to} \;\; s \geq 0. \tag{4.34}$$

The first-order KKT conditions for problem (4.34) are

$$J(x^*)^{\mathrm{T}}\big(c(x^*) - s^*\big) = 0, \qquad\qquad s^* \geq 0, \tag{4.35a}$$

$$s^* \cdot \big(c(x^*) - s^*\big) = 0, \qquad c(x^*) - s^* \leq 0. \tag{4.35b}$$

These conditions define an infeasible stationary point.

Definition 4.5.2 (Infeasible stationary point)**.** *The pair* (x^*, s^*) *is an* infeasible stationary point *if* $c(x^*) - s^* \neq 0$ *and* (x^*, s^*) *satisfies the optimality conditions* (4.35)*.*

Lemma 4.5.2. *If* $|\mathcal{O}| < \infty$*, then* $|\mathcal{M}| = \infty$*.*

Proof. The proof is by contradiction. Suppose that $|\mathcal{M}| < \infty$, in which case $|\mathcal{O} \cup \mathcal{M}| < \infty$. It follows from the definition of Algorithm 5 that $k \in \mathcal{F}$ for all k sufficiently large, i.e., there must exist an iteration index k_F such that

$$k \in \mathcal{F}, \;\; y_k^E = y^E, \;\; \text{and} \;\; (\tau_k, w_k^E, \mu_k^P, \mu_k^B) = (\tau, w^E, \mu^P, \mu^B) > 0 \tag{4.36}$$

for all $k \geq k_F$. The updating rule for $\{\mu_k^L\}$ implies that μ_k^L will be fixed at some $\mu^L \geq \mu^P$, and μ_k^F is then fixed at the value μ^L for all k sufficiently large. It follows from Theorem 4.4.1 that there exists a subsequence of iterates $\mathcal{S}$ such that

$$\lim_{k \to \mathcal{S}} \|\nabla M(v_k)\| = 0.$$

Then Lemma 4.4.2(i) and Lemma 4.4.2(ii) can be applied to show that (4.25) is satisfied for all $k \in \mathcal{S}$. This would mean, in view of Step 13 of Algorithm 5, that $\mathcal{S} \in \mathcal{M}$ with $|\mathcal{S}| = \infty$, which contradicts (4.36) because $\mathcal{F} \cap \mathcal{M} = \emptyset$. □

For the next lemma, we introduce the quantities

$$\pi_{k+1}^Y = y_k^E - \frac{1}{\mu_k^P}\big(c(x_{k+1}) - s_{k+1}\big) \;\; \text{and} \;\; \pi_{k+1}^W = \mu_k^B (S_{k+1} + \mu_k^B I)^{-1}(w_k^E - s + s_k^E) \tag{4.37}$$

with $S_{k+1} = \operatorname{diag}(s_{k+1})$ associated with the gradient of the merit function in (4.7).

Lemma 4.5.3. *If* $|\mathcal{M}| = \infty$ *then*

$$\lim_{k \in \mathcal{M}} \|\pi_{k+1}^Y - y_{k+1}\| = 0.$$

Moreover, if there exists a subsequence of iterates $\mathcal{K} \subseteq \mathcal{M}$ *such that* $\lim_{k \in \mathcal{K}} \; s_k = s^* \geq 0$*, then*

$$\lim_{k \in \mathcal{K}} \; \|\pi_{k+1}^W - w_{k+1}\| = \lim_{k \in \mathcal{K}} \; \|\pi_{k+1}^Y - \pi_{k+1}^W\| = \lim_{k \in \mathcal{K}} \; \|y_{k+1} - w_{k+1}\| = 0.$$

Proof. It follows from (4.7) and (4.25c) that

$$\|\pi_{k+1}^Y - y_{k+1}\| \leq \tau_k. \tag{4.38}$$

As $|\mathcal{M}| = \infty$ by assumption, Step 14 of Algorithm 5 implies that $\lim_{k\to\infty}\ \tau_k = 0$. Combining this with (4.38) establishes the first limit in the result.

Furthermore, if there exists a subsequence $\mathcal{K} \subseteq \mathcal{M}$ such that $\lim_{k\in\mathcal{K}}\ s_k = s^* \geq 0$, then the updating rule of Algorithm 5 for $s_k^{\scriptscriptstyle E}$ implies that $\lim_{k\in\mathcal{K}}\ (s_k^{\scriptscriptstyle E} - s_k) = 0$. The limit $\lim_{k\to\infty}\ \tau_k = 0$ may then be combined with (4.7), (4.25b) and (4.25c) to show that

$$\lim_{k\in\mathcal{K}}\ \|\pi_{k+1}^{\scriptscriptstyle W} - w_{k+1}\| = 0 \quad \text{and} \quad \lim_{k\in\mathcal{K}}\ \|\pi_{k+1}^{\scriptscriptstyle Y} - \pi_{k+1}^{\scriptscriptstyle W}\| = 0. \tag{4.39}$$

Finally, as $\lim_{k\to\infty}\ \tau_k = 0$, it follows from the bound (4.38) and limits (4.39) that

$$\lim_{k\in\mathcal{K}}\ \|y_{k+1} - w_{k+1}\| = \lim_{k\in\mathcal{K}}\ \|(y_{k+1} - \pi_{k+1}^{\scriptscriptstyle Y}) + (\pi_{k+1}^{\scriptscriptstyle Y} - \pi_{k+1}^{\scriptscriptstyle W}) + (\pi_{k+1}^{\scriptscriptstyle W} - w_{k+1})\| = 0.$$

This establishes the last of the four limits. □

Lemma 4.5.4. *If $|\mathcal{O}| < \infty$, then every limit point (x^*, s^*) of the subsequence $\{\, (x_{k+1}, s_{k+1})\,\}_{k\in\mathcal{M}}$ satisfies $c(x^*) - s^* \neq 0$.*

Proof. The proof is similar to the proof of Lemma 4.7 in [46] but with some modified technical details.

Let (x^*, s^*) be a limit point of (the necessarily infinite) sequence $\mathcal{M}$, i.e., there exists a subsequence $\mathcal{K} \subseteq \mathcal{M}$ such that $\lim_{k\in\mathcal{K}}\ (x_{k+1}, s_{k+1}) = (x^*, s^*)$. For a proof by contradiction, assume that $c(x^*) - s^* = 0$, which implies that

$$\lim_{k\in\mathcal{K}}\ \|c(x_{k+1}) - s_{k+1}\| = 0. \tag{4.40}$$

First, we show that $s^* \geq 0$, which will imply that $(x^*,\ s^*)$ is feasible because of the assumption that $c(x^*) - s^* = 0$. The line search in Algorithm 4 gives $s_{k+1} + \mu_k^{\scriptscriptstyle B} e > 0$ for all k. If $\lim_{k\to\infty}\ \mu_k^{\scriptscriptstyle B} = 0$, then $s^* = \lim_{k\in\mathcal{K}}\ s_{k+1} \geq -\lim_{k\in\mathcal{K}} \mu_k^{\scriptscriptstyle B} e = 0$. On the other hand, if $\lim_{k\to\infty}\ \mu_k^{\scriptscriptstyle B} \neq 0$, then Step 19 of Algorithm 5 is executed a finite number of times, $\mu_k^{\scriptscriptstyle B} = \mu^{\scriptscriptstyle B} > 0$ and (4.28) holds for all $k \in \mathcal{M}$ sufficiently large. A combination of the assumption that $|\mathcal{O}| < \infty$, the result of Lemma 4.5.2, and the updates of Algorithm 5, establishes that $\lim_{k\to\infty}\ \tau_k = 0$ and

$$\chi_k^{\max} = \chi^{\max} > 0 \ \text{ for all sufficiently large } k \in \mathcal{K}. \tag{4.41}$$

Taking limits over $k \in \mathcal{M}$ in (4.28) and using $\lim_{k\to\infty}\ \tau_k = 0$ gives $s^* \geq 0$.

Using $|\mathcal{O}| < \infty$ together with Lemma 4.5.3, the fact that $\lim_{k\in\mathcal{K}}\ s_k = s^* \geq 0$ with $\mathcal{K} \subseteq \mathcal{M}$, and Step 16 of the line search of Algorithm 4 gives

$$\lim_{k\in\mathcal{K}} \|y_{k+1} - w_{k+1}\| = 0, \text{ and } w_{k+1} + \mu^{B}_{k+1} > 0 \text{ for all } k \geq 0. \tag{4.42}$$

Next, it can be observed from the definitions of π^{Y}_{k+1} and $\nabla_x M$ that

$$\begin{aligned}\nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}} y_{k+1} &= \nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}}(2\pi^{Y}_{k+1} + y_{k+1} - 2\pi^{Y}_{k+1}) \\ &= \nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}}\big(2\pi^{Y}_{k+1} - y_{k+1}\big) - 2J(x_{k+1})^{\mathrm{T}}(y_{k+1} - \pi^{Y}_{k+1}) \\ &= \nabla_x M(v_{k+1}\,; y^{E}_{k}, w^{E}_{k}, \mu^{P}_{k}, \mu^{B}_{k}) - 2J(x_{k+1})^{\mathrm{T}}(y_{k+1} - \pi^{Y}_{k+1}),\end{aligned}$$

which combined with $\{\, x_{k+1} \,\}_{k\in\mathcal{K}} \to x^*$, $\lim_{k\to\infty} \tau_k = 0$, (4.25a), and Lemma 4.5.3 gives

$$\lim_{k\in\mathcal{K}} \{\, \nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}} y_{k+1} \,\} = 0. \tag{4.43}$$

The proof that $\lim_{k\in\mathcal{K}} \chi_{\mathrm{comp}}(v_{k+1}, \mu^{B}_{k}) = 0$ involves two cases.

Case 1: $\lim_{k\to\infty} \mu^{B}_{k} \neq 0$. In this case $\mu^{B}_{k} = \mu^{B} > 0$ for all sufficiently large k. Combining this with $|\mathcal{M}| = \infty$ and the update to μ^{B}_{k} in Step 19 of Algorithm 5, it must be that (4.28) holds for all sufficiently large $k \in \mathcal{K}$, i.e., that $\chi_{\mathrm{comp}}(v_{k+1}, \mu^{B}_{k}) \leq \tau_k$ for all sufficiently large $k \in \mathcal{K}$. As $\lim_{k\to\infty} \tau_k = 0$, it must hold that $\lim_{k\in\mathcal{K}} \chi_{\mathrm{comp}}(v_{k+1}, \mu^{B}_{k}) = 0$.

Case 2: $\lim_{k\to\infty} \mu^{B}_{k} = 0$. Lemma 4.5.3 implies that $\lim_{k\in\mathcal{K}} (\pi^{W}_{k+1} - w_{k+1}) = 0$. The sequence $\{\, S_{k+1} + \mu^{B}_{k} I \,\}_{k\in\mathcal{K}}$ is bounded because $\{\, \mu^{B}_{k} \,\}$ is positive and monotonically decreasing and $\lim_{k\in\mathcal{K}} s_{k+1} = s^*$, which means by the definition of π^{W}_{k+1} and the updating rule for s^{E}_{k+1} in (4.26),

$$0 = \lim_{k\in\mathcal{K}} (S_{k+1} + \mu^{B}_{k} I)(\pi^{W}_{k+1} - w_{k+1}) = \lim_{k\in\mathcal{K}} \big(\mu^{B}_{k} w^{E}_{k} - (S_{k+1} + \mu^{B}_{k} I) w_{k+1}\big). \tag{4.44}$$

Moreover, as $|\mathcal{O}| < \infty$ and $w_k > 0$ for all k by construction, the updating strategy for w^{E}_{k} in Algorithm 5 guarantees that $\{\, w^{E}_{k} \,\}$ is bounded over all k (see (4.26)). It then follows from (4.44), the uniform boundedness of $\{\, w^{E}_{k} \,\}$, and $\lim_{k\to\infty} \mu^{B}_{k} = 0$ that

$$0 = \lim_{k\in\mathcal{K}} \big([s_{k+1}]_i + \mu^{B}_{k}\big)[w_{k+1}]_i = \lim_{k\in\mathcal{K}} \big([s_{k+1}]_i + \mu^{B}_{k}\big)([w_{k+1}]_i + \mu^{B}_{k}). \tag{4.45}$$

There are two subcases.

Subcase 2a: $s^*_i > 0$ for some i. As $\lim_{k\in\mathcal{K}} [\, s_{k+1} \,]_i = s^*_i > 0$ and $\lim_{k\to\infty} \mu^{B}_{k} = 0$, it follows from (4.45) that $\lim_{k\in\mathcal{K}} [\, w_{k+1} \,]_i = 0$. Combining these limits allows us to conclude that $\lim_{k\in\mathcal{K}} [\, q_1(v_{k+1}) \,]_i = 0$, which is the desired result for this case.

Subcase 2b: $s_i^* = 0$ for some i. In this case, it follows from the limits $\lim_{k\to\infty}\mu_k^B = 0$ and (4.45), $w_{k+1} + \mu_k^B > 0$ and the limit $\lim_{k\in\mathcal{K}}[\, s_{k+1}\,]_i = s_i^* = 0$ that $\lim_{k\in\mathcal{K}}[q_2(v_{k+1}, \mu_k^B)]_i = 0$, which is the desired result for this case.

As one of the two subcases above must occur for each component i, it follows that

$$\lim_{k\in\mathcal{K}} \chi_{\text{comp}}(v_{k+1}, \mu_k^B) = 0,$$

which completes the proof for Case 2.

Under the assumption $c(x^*) - s^* = 0$ it has been shown that (4.40), (4.42), (4.43), and the limit $\lim_{k\in\mathcal{K}}\ \chi_{\text{comp}}(v_{k+1}, \mu_k^B) = 0$ hold. Collectively, these results imply that $\lim_{k\in\mathcal{K}} \chi(v_{k+1}, \mu_k^B) = 0$. This limit, together with the inequality (4.41) and the condition checked in Step 10 of Algorithm 5, gives $k \in \mathcal{O}$ for all $k \in \mathcal{K} \subseteq \mathcal{M}$ sufficiently large. This is a contradiction because $\mathcal{O}\cap\mathcal{M} = \emptyset$, which establishes the desired result that $c(x^*) - s^* \neq 0$. □

Lemma 4.5.5. *If $|\mathcal{O}| < \infty$, then there exists at least one limit point (x^*, s^*) of the infinite sequence $\{\, (x_{k+1}, s_{k+1})\,\}_{k\in\mathcal{M}}$, and any such limit point is an infeasible stationary point as given by* Definition 4.5.2.

Proof. The proof is similar to the proof of Lemma 4.8 in [46] but with some modified technical details.

If $|\mathcal{O}| < \infty$ then Lemma 4.5.2 implies that $|\mathcal{M}| = \infty$. Moreover, the updating strategy of Algorithm 5 forces $\{\, y_k^E \,\}$ and $\{\, w_k^E \,\}$ to be bounded (see (4.26)). The next step is to show that $\{\, s_{k+1}\,\}_{k\in\mathcal{M}}$ is bounded.

For a proof by contradiction, suppose that $\{\, s_{k+1}\,\}_{k\in\mathcal{M}}$ is unbounded. It follows that there must be a component i and a subsequence $\mathcal{K} \subseteq \mathcal{M}$ for which $\{\, [\, s_{k+1}\,]_i\,\}_{k\in\mathcal{K}} \to \infty$. When Assumption 4.4.3 and Assumption 4.4.1 hold, $\{\, c(x_{k+1})\,\}_{k\in\mathcal{K}}$, $\{\, \nabla f(x_{k+1})\,\}_{k\in\mathcal{K}}$ and $\{\, J(x_{k+1})\,\}_{k\in\mathcal{K}}$ must be bounded. This implies that $\{\, [\, \pi_{k+1}^Y\,]_i\,\}_{k\in\mathcal{K}}$ is unbounded. On the other hand, by (4.7), (4.25a), together with the limit $\lim_{k\to\infty}\tau_k = 0$ and Lemma 4.5.3,

$$\begin{aligned}
0 &= \lim_{k\in\mathcal{M}} \|\nabla_x M(v_{k+1}\,; y_k^E, w_k^E, \mu_k^P, \mu_k^B)\| \\
&= \lim_{k\in\mathcal{M}} \|\nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}}\pi_{k+1}^Y - J(x_{k+1})^{\mathrm{T}}(\pi_{k+1}^Y - y_{k+1})\| \\
&= \lim_{k\in\mathcal{M}} \|\nabla f(x_{k+1}) - J(x_{k+1})^{\mathrm{T}}\pi_{k+1}^Y\| = 0,
\end{aligned}$$

which contradicts the unboundedness of $\{\, [\, \pi_{k+1}^Y\,]_i\,\}_{k\in\mathcal{K}}$. Thus, it must be the case that $\{\, s_{k+1}\,\}_{k\in\mathcal{M}}$ is bounded.

The next part of the proof is to establish that $s^* \geq 0$, which is the inequality condition of (4.35a). The test in Step 16 of Algorithm 5 (i.e., testing whether (4.28) holds) is checked infinitely often because $|\mathcal{M}| = \infty$. If (4.28) is satisfied finitely many times, then the update $\mu^B_{k+1} = \frac{1}{2}\mu^B_k$ forces $\{\, \mu^B_{k+1} \,\} \to 0$. Combining this with $s_{k+1} + \mu^B_k e > 0$ shows that $s^* \geq 0$, as claimed. On the other hand, if (4.28) is satisfied for all sufficiently large $k \in \mathcal{M}$, then $\mu^B_{k+1} = \mu^B > 0$ for all sufficiently large k and $\lim_{k\in\mathcal{K}} \chi_{\text{comp}}(v_{k+1}, \mu^B_k) = 0$ because $\{\, \tau_k \,\} \to 0$. It follows from these two facts that $s^* \geq 0$, as claimed.

The boundedness of $\{\, s_{k+1} \,\}_{k\in\mathcal{M}}$ and Assumption 4.4.3 ensure the existence of at least one limit point of $\{\, (x_{k+1}, s_{k+1}) \,\}_{k\in\mathcal{M}}$. If (x^*, s^*) is any such limit point, there must be a subsequence $\mathcal{K} \subseteq \mathcal{M}$ such that $\{\, (x_{k+1}, s_{k+1}) \,\}_{k\in\mathcal{K}} \to (x^*, s^*)$. It remains to show that (x^*, s^*) is an infeasible stationary point (i.e., that (x^*, s^*) satisfies the optimality conditions (4.35a)–(4.35b)).

As $|\mathcal{O}| < \infty$, it follows from Lemma 4.5.4 that $c(x^*) - s^* \neq 0$. Combining this with $\{\, \tau_k \,\} \to 0$, which holds because $\mathcal{K} \subseteq \mathcal{M}$ is infinite (on such iterations $\tau_{k+1} \leftarrow \frac{1}{2}\tau_k$), it follows that the condition (4.27) of Step 15 of Algorithm 5 will not hold for all sufficiently large $k \in \mathcal{K} \subseteq \mathcal{M}$. The subsequent updates ensure that $\{\, \mu^P_k \,\} \to 0$, hence $\{\, \mu^F_k \,\} \to 0$ by the updating rule for $\{\, \mu^L_k \,\}$, which, combined with (4.17), the boundedness of $\{\, y^E_k \,\}$, and Lemma 4.5.3, gives

$$\{c(x_{k+1}) - s_{k+1}\}_{k\in\mathcal{K}} \leq \big\{\mu^F_k\big(y^E_k + \tfrac{1}{2}(w_{k+1} - y_{k+1}) + \mu^B_k\big)\big\}_{k\in\mathcal{K}} \to 0.$$

This implies that $c(x^*) - s^* \leq 0$ and the second condition in (4.35b) holds.

For a proof of the equality condition of (4.35a) observe that the gradients must satisfy $\{\nabla_{\!x} M(v_{k+1}\,; y^E_k, w^E_k, \mu^P_k, \mu^B_k)\}_{k\in\mathcal{K}} \to 0$ because condition (4.25) is satisfied for all $k \in \mathcal{M}$ (cf. Step 13 of Algorithm 5). Multiplying $\nabla_{\!x} M(v_{k+1}\,; y^E_k, w^E_k, \mu^P_k, \mu^B_k)$ by μ^P_k, and applying the definition of π^Y_{k+1} from (4.37) yields

$$\big\{\mu^P_k g(x_{k+1}) - J(x_{k+1})^T\big(\mu^P_k \pi^Y_{k+1} + \mu^P_k(\pi^Y_{k+1} - y_{k+1})\big)\big\}_{k\in\mathcal{K}} \to 0.$$

Combining this with $\{x_{k+1}\}_{k\in\mathcal{K}} \to x^*$, $\{\mu^P_k\} \to 0$, and the result of Lemma 4.5.3 yields

$$\big\{-J(x_{k+1})^T(\mu^P_k \pi^Y_{k+1})\big\}_{k\in\mathcal{K}} = \big\{-J(x_{k+1})^T(\mu^P_k y^E_k - c(x_{k+1}) + s_{k+1})\big\}_{k\in\mathcal{K}} \to 0.$$

Using this limit in conjunction with the boundedness of $\{y^E_k\}$, the fact that $\{\mu^P_k\} \to 0$, and $\{(x_{k+1}, s_{k+1}\}_{k\in\mathcal{K}} \to (x^*, s^*)$ establishes that the first condition of (4.35a) holds.

It remains to show that the complementarity condition of (4.35b) holds. From Lemma 4.5.3 it must be the case that $\{\pi^W_{k+1} - \pi^Y_{k+1}\}_{k\in\mathcal{K}} \to 0$. Also, the limiting value does not change if the sequence is multiplied (term by term) by the bounded sequence $\{\mu_k^P(S_{k+1}+\mu_k^B I)\}_{k\in\mathcal{K}}$ (recall that $\{s_{k+1}\}_{k\in\mathcal{K}}\to s^*$). This yields

$$\big\{\mu_k^B\mu_k^P(w_k^E - s_{k+1} + s_k^E) - \mu_k^P(S_{k+1}+\mu_k^B I)y_k^E + (S_{k+1}+\mu_k^B I)(c(x_{k+1}) - s_{k+1})\big\}_{k\in\mathcal{K}} \to 0.$$

This limit, together with the limits $\{\mu_k^P\}\to 0$ and $\{s_{k+1}\}_{k\in\mathcal{K}}\to s^*$, and the boundedness of $\{y_k^E\}$ and $\{w_k^E\}$ implies that

$$\big\{(S_{k+1}+\mu_k^B I)(c(x_{k+1}) - s_{k+1})\big\}_{k\in\mathcal{K}} \to 0. \tag{4.46}$$

As $c(x^*) - s^* \neq 0$, there must exist a constraint index i such that $[\,c(x^*) - s^*\,]_i \neq 0$. Combining this with $\{(x_{k+1}, s_{k+1})\}_{k\in\mathcal{K}} \to (x^*, s^*)$ and (4.46) shows that $\{[\,s_{k+1}\,]_i + \mu_k^B\}_{k\in\mathcal{K}} \to 0$. As s^* is nonnegative, it follows that $\{\mu_k^B\}_{k\in\mathcal{K}}\to 0$, However, as $\{\mu_k^B\}$ is a monotonically decreasing sequence, it must hold that $\{\mu_k^B\}\to 0$. Using this fact, (4.46), and $\{(x_{k+1}, s_{k+1})\}_{k\in\mathcal{K}} \to (x^*, s^*)$ it follows that $s^*\cdot\big(c(x^*) - s^*\big) = 0$, and the first condition in (4.35b) holds. This completes the proof. □

Theorem 4.5.2. *Under* Assumptions 4.4.1–4.4.3*, one of the following occurs:*

(i) $|\mathcal{O}| = \infty$*, limit points of* $\{\,(x_{k+1}, s_{k+1})\,\}_{k\in\mathcal{O}}$ *exist, and every such limit point* (x^*, s^*) *is a* CAKKT point for problem (NIPs). If, in addition, CAKKT-regularity holds at (x^*, s^*), then (x^*, s^*) is a KKT point for problem (NIPs).

(ii) $|\mathcal{O}| < \infty$, $|\mathcal{M}| = \infty$, limit points of $\{\,(x_{k+1}, s_{k+1})\,\}_{k\in\mathcal{M}}$ exist, and every such limit point (x^*, s^*) is an infeasible stationary point.

Proof. Part (i) follows from Lemma 4.5.1 and Theorem 4.5.1. Part (ii) follows from Lemma 4.5.5. Also, the exclusive conditions on $|\mathcal{O}|$ imply that only one of these two cases must occur. □

Chapter 4 and the numerical results in Section 5.3, in part, reprint the paper by Philip E. Gill and Minxin Zhang, "A projected-search interior method for nonlinear optimization." Manuscript submitted for publication, 2023. The dissertation author was the primary investigator and author of the paper.

Chapter 5

Numerical Results

5.1 A Projected-Search Active-Set Method for Bound Constraints

5.1.1 The implementation

Numerical results were obtained for the projected-search active-set method UBOPT, in which the direction d_k was computed as the solution of (3.21) with H_k chosen as a positive-definite limited-memory BFGS approximation of $\nabla^2 f(x_k)$. All testing was done on problems taken from the CUTEst test collection (see Bongartz, Conn, Gould and Toint [7] and Gould, Orban and Toint [58]). The CUTEst test set contains 154 bound-constrained problems of the form (BC). Although many problems allow for the number of variables and constraints to be adjusted in the standard interface format (SIF) data file, our tests used the default dimensions set in the CUTEst distribution. This gave problems ranging in size from `BQ1VAR` (one variable) to `WALL100` (149624 variables).

The practical effectiveness of the quasi-Wolfe search was evaluated by running two limited-memory quasi-Newton methods, one with a quasi-Wolfe search and the other with a quasi-Armijo search. The resulting implementations, `UBOPT-qWolfe` and `UBOPT-qArmijo`, are based on the Fortran package UBOPT (see Ferry, Gill, Wong and Zhang [30]). In the quasi-Wolfe search, the kink steps are sorted in decreasing order in $O(n \log n)$ flops using a heapsort algorithm (see, e.g., Williams [85], Knuth [65, Section 5.2.3]), adapted from a Fortran implementation by Byrd, Lu, Nocedal and Zhu [10]. For `UBOPT-qWolfe`, the Armijo tolerance η_A was set at 10^{-4} and the Wolfe tolerance $\eta_W = 0.9$. In `UBOPT-qArmijo`, $\eta_A = 0.3$. The scalar ϵ was set to the machine precision in the expression for ϵ_k in the calculation (3.5) of the working set.

In order to provide some measure of the efficiency of the projected-search method relative to a state-of-the-art method for bound-constrained optimization, the solvers `UBOPT-qWolfe` and `UBOPT-qArmijo` were compared with the limited-memory method `LBFGS-B` (Byrd, Lu, Nocedal and Zhu [10], Zhu, Byrd, Lu and Nocedal[95], and Morales and Nocedal [68]). All three solvers were applied to the 154 bound-constrained problems from the CUTEst test set. The runs were terminated at the first point x_T such that

(**a**) $\|P_{x_T}\big(-\nabla f(x_T)\big)\|_\infty \le 10^{-5}\big(1+|f(x_T)|\big)$ and

(**b**) $|f(x_T)-f(x_{T-1})| \le 10^7\epsilon_M \times \max\big\{\,|f(x_T)|, |f(x_{T-1})|, 1\,\big\}$; or

(**c**) $\|P_{x_T}\big(-\nabla f(x_T)\big)\|_\infty < \sqrt{\epsilon_M}$,

where ϵ_M is the machine precision. In the first iteration of the algorithms, only condition ((**c**)) is tested. A nonoptimal termination was signaled by the violation of a time limit of 3600 seconds, a limit of 10^6 iterations, or an abnormal exit because of numerical difficulties.

5.1.2 Numerical results

The solver `UBOPT-qArmijo` failed on nine problems, with six failing because of numerical difficulties (`BLEACHNG`, `BQPGAUSS`, `BRATU1D`, `GRIDGENA`, `RAYBENDL`, `WALL10`, and `WEEDS`). The solver `UBOPT-qWolfe` failed on six problems, with four failures caused by numerical difficulties (`GRIDGENA`, `PALMER5E`, `PROBPENL`, and `WALL10`). `UBOPT-qWolfe` identified problem `BRATU1D` as being unbounded. For both solvers, `CYCLOOCTLS` and `WALL50` could not be solved within the one hour time limit. In the cases of numerical difficulties, the search algorithms were unable to compute an appropriate step. We note that for `UBOPT-qWolfe`, the run for `PROBPENL` terminated at a near-optimal point that satisfied condition ((**a**)) and $\|P_{x_T}\big(-\nabla f(x_T)\big)\|_\infty = 1.99\times 10^{-7}$. The solver `LBFGS-B` failed on 16 problems. Seven failures were caused by numerical difficulties (`BQPGAUSS`, `BRATU1D`, `GRIDGENA`, `PALMER5A`, `PALMER5B`, `PALMER7A`, and `WALL10`), seven problems exceeded the iteration limit (`CHEBYQAD`, `PALMER1E`, `PALMER2E`, `PALMER3E`, `PALMER4E`, `PALMER6E`, and `PALMER8E`), and two problems exceeded the time limit (`CYCLOOCTLS` and `WALL50`).

The relative performance of the solvers is summarized using performance profiles (in $\log_2$ scale), which were proposed by Dolan and Moré [24]. Let $\mathcal{P}$ denote a set of problems used for a given numerical experiment. For each method s we define the function $\pi_s : [0, r_M] \mapsto \mathbb{R}^+$ such that

$$\pi_s(\tau) = \frac{1}{n_p}\left|\big\{\,p\in\mathcal{P} : \log_2(r_{p,s}) \le \tau\,\big\}\right|,$$

where n_p is the number of problems in the test set and $r_{p,s}$ denotes the ratio of the number of function evaluations needed to solve problem p with method s and the least number of function evaluations needed to solve problem p. If method s failed for problem p, then $r_{p,s}$ is set to be twice of the maximal ratio. The parameter r_M is the maximum value of $\log_2(r_{p,s})$.

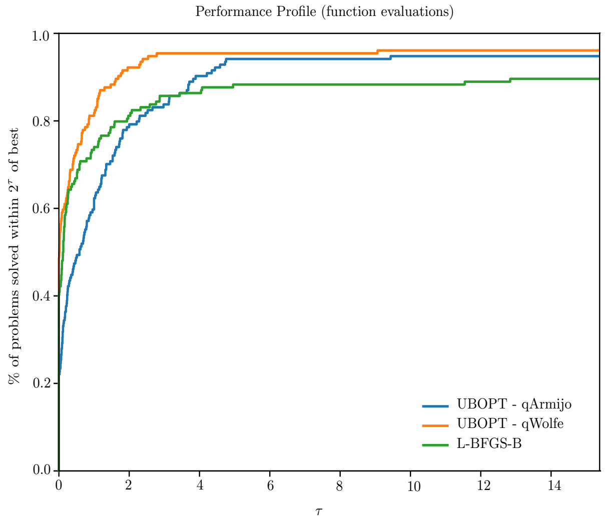

Figure 5.1: Performance profiles for the number of function evaluations required to solve 154 bound-constrained problems from the CUTEst test set. The figure gives the profiles for the three solvers `UBOPT-qWolfe`, `UBOPT-qArmijo`, and `L-BFGS-B` [10].

Figure 5.1 gives the function-evaluation performance profiles for the 154 bound-constrained problems for `UBOPT-qWolfe`, `UBOPT-qArmijo`, and `LBFGS-B`. The profile utilized the total number of function evaluations for comparison. Additional information about the runs used to generate the performance profiles is given by Ferry, Gill, Wong and Zhang [29]. The results indicate that using a quasi-Wolfe search in UBOPT resulted in a substantially better performance with respect to function calls than using a quasi-Armijo search, and comparable and more robust performance with respect to `LBFGS-B`.

A benefit of the Wolfe conditions in the unconstrained case is that the restriction on the directional derivative guarantees that the approximate curvature $\big(\nabla f(x_{k+1}) - \nabla f(x_k)\big)^{\mathrm{T}}(x_{k+1} - x_k)$ is positive, which is a necessary condition for the quasi-Newton update to give a positive-definite approximate Hessian. In the bound-constrained case, the use of a quasi-Wolfe projected search makes it more likely that the update can be applied, but it is not possible to guarantee an update in all cases. If the next iterate is given by $x_{k+1} = \mathbf{proj}_{\Omega}(x_k + \alpha_k p_k)$, where α_k is a quasi-Wolfe step, then $\big(\nabla f(x_{k+1}) - \nabla f(x_k)\big)^{\mathrm{T}}(x_{k+1} - x_k)$ need not be greater than zero if the path $\mathbf{proj}_{\Omega}(x_k + \alpha_k p_k)$ changes direction for some $\alpha \in (0, \alpha_k)$. If it does change direction, $\psi'_+(0)$ and $\psi'_-(\alpha_k)$ may be directional derivatives of f in a direction other than $x_{k+1} - x_k$. This situation is illustrated in Figure 5.2, which depicts a two-dimensional region with lower bounds $x_1 = 0$ and $x_2 = 0$. In this example $\psi'_+(0)$ is a directional derivative of f in direction $[\,p_k\,]_1$ and $\psi'_-(\alpha_k)$ is a directional derivative of f in direction $[\,p_k\,]_2$. As a result, if the path changes direction for $\alpha \in (0, \alpha_k)$, then there is the possibility that the quasi-Newton update must be skipped.

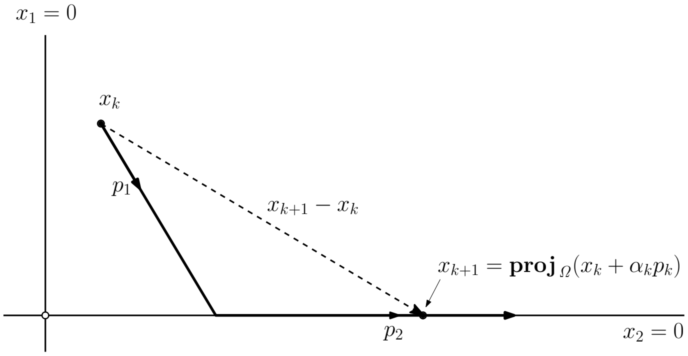

Figure 5.2: Example with no guarantee of an update for the approximate Hessian.

It is shown in Section 3.3.2 that if $\{\,x_k\,\}$ converges to a nondegenerate stationary point, then a quasi-Wolfe search identifies the active set at the solution in a finite number of iterations. After the active set stabilizes, a quasi-Wolfe search behaves exactly like a Wolfe line search in the sense that updates to the approximate Hessian are guaranteed if $f(x_k + \alpha p_k)$ is bounded below.

To estimate how often the update is likely to be skipped with the quasi-Wolfe search, statistics were collected from the test problems for which at least one of the search paths was “bent” by projection. In total, the application of `UBOPT-qWolfe` resulted in 259 of the potential 637268 updates being skipped ($\approx$ 0.04%). This can be compared to 6537 of the 679071 updates being skipped ($\approx$ 1.0%) for `UBOPT-qArmijo`. (The number of updates reflects the number of iterations needed for convergence.)

5.2 A Projected-Search Interior Method for Bound Constraints

5.2.1 The implementation

Numerical results are given for a MATLAB implementation of a projected-search method based on the primal-dual interior method of Forsgren and Gill [34]. Applying this method to bound-constrained problems, the unconstrained function

$$\begin{aligned} M(x, z_1, z_2\,; \mu) \triangleq f(x) &- \sum_{j=1}^{n} \big\{\mu \ln\big(x_j - \ell_j\big) + \mu \ln\big([z_1]_j(x_j - \ell_j)\big) - [z_1]_j(x_j - \ell_j)\big\} \\ &- \sum_{j=1}^{n} \big\{\mu \ln\big(u_j - x_j\big) + \mu \ln\big([z_2]_j(u_j - x_j)\big) - [z_2]_j(u_j - x_j)\big\} \end{aligned}$$

is minimized for a sequence of μ-values such that $\mu \to 0$. This implies that the function M of problem (IPBC) is given by $M(v\,;\mu) = M(x, z_1, z_2\,;\mu)$, with

$$v = \begin{pmatrix} x \\ z_1 \\ z_2 \end{pmatrix}, \qquad \ell_v = \begin{pmatrix} \ell \\ 0 \\ 0 \end{pmatrix}, \quad \text{and} \quad u_v = \begin{pmatrix} u \\ +\infty \\ +\infty \end{pmatrix}.$$

At any (x, z_1, z_2) such that $\ell < x < u$, $z_1 > 0$ and $z_2 > 0$, let $X_1 = \operatorname{diag}\big(x_j - \ell_j\big)$, $X_2 = \operatorname{diag}\big(u_j - x_j\big)$, $Z_1 = \operatorname{diag}([z_1]_j)$, and $Z_2 = \operatorname{diag}([z_2]_j)$. One iteration of Newton's method for minimizing $M(x, z_1, z_2\,;\mu)$ requires solving the equations $\nabla^2 M(v\,;\mu)\Delta v = -\nabla M(v\,;\mu)$. If the diagonal matrices μX_1^{-1} and μX_2^{-1} in the expression for $\nabla^2 M(v\,;\mu)$ are replaced by Z_1 and Z_2, we obtain an approximate Hessian with $n \times n$ principal minor $H_k = \nabla^2 f(x) + X_1^{-1} Z_1 + X_2^{-1} Z_2$. It follows that one iteration of an *approximate* Newton method for minimizing $M(x, z_1, z_2\,;\mu)$ gives the estimate $(x + \Delta x,\ z_1 + \Delta z_1,\ z_2 + \Delta z_2)$, where $\Delta z_1 = -X_1^{-1}\big(z_1 \cdot (x + \Delta x - \ell) - \mu e\big)$, $\Delta z_2 = -X_2^{-1}\big(z_2 \cdot (u - x - \Delta x) - \mu e\big)$, and Δx satisfies the equations

$$H_k \Delta x = -\big(\nabla f(x) - \mu X_1^{-1} e + \mu X_2^{-1} e\big). \tag{5.1}$$

Let v_k denote a point such that $\ell_v < v_k < u_v$ and let Δv_k denote the solution of the approximate Newton equations at v_k. If the matrix H_k of (5.1) is positive definite, then Δv_k is a descent direction for $M(x, z_1, z_2\,;\mu)$. Otherwise a positive-definite modified matrix $\widehat{H}_k \approx H_k$ must be used. If necessary, the matrix H_k was modified using the method of Wächter and Biegler [84, Algorithm IC, p. 36], which factors the matrix $H_k + \delta I_n$ for some $\delta > 0$. Each (possibly perturbed)

H_k matrix was factored using the MATLAB built-in command LDL, which uses the routine MA57 [25].

5.2.2 Numerical results

Results are presented from two variants of the Forsgren-Gill method. The first variant, `PD-Wolfe`, is the conventional primal-dual method implemented with a Wolfe line search; the second variant, `PDproj-qWolfe`, is the projected-search interior method proposed in Section 3.5. As the underlying interior method is the same in both cases, the results show the benefits of formulating the method as a projected-search method.

The algorithms were considered to have solved a problem successfully if

$$\max\Big\{\|\max(0, g(x) \cdot x_\ell)\|_\infty, \|\max(0, -g(x) \cdot x_u)\|_\infty\Big\} \le 10^{-5},$$

where $x_\ell = \min\{1, (x-\ell) \mathbin{\cdot/} (1+|\ell|)\}$, $x_u = \min\{1, (u-x) \mathbin{\cdot/} (1+|u|)\}$, and

$$g(x) = \nabla f(x)/(\max\{1, \|\nabla f(x)\|_\infty\}).$$

A limit of 500 was placed on the number of iterations. The strategy for choosing the barrier parameter μ was that used in the method of Gertz and Gill [44]. The fraction-to-the-boundary parameter σ of (4.15) was set at 0.9.

All testing was done using MATLAB version R2019a on an iMac with a 3.0 GHz Intel Xeon W processor and 128 GB of 800 MHz DDR4 RAM running macOS, version 10.14.6 (64 bit). Results were obtained for a subset of the bound-constrained problems in CUTEst for which the dimension of the problem n is 600 or less, or n may be set at the largest value less than 1000. This gave 137 problems ranging in size from `BQP1VAR` (one variable) to `POWELLBC` (1000 variables). Exact second derivatives were used for all the runs.

Figure 5.3 gives the performance profiles for the total number of iterations as well as the total number of function evaluations required to solve the 137 problems. The profiles compare the primal-dual interior method `PD-Wolfe` implemented with a Wolfe line search and a projected-search interior method `PDproj-qWolfe` with a quasi-Wolfe line search (i.e., the method described in Section 3.5). Figure 5.3 indicates that a projected-search interior method with a quasi-Wolfe line search can provide substantial improvements in robustness and performance compared to a conventional interior method.

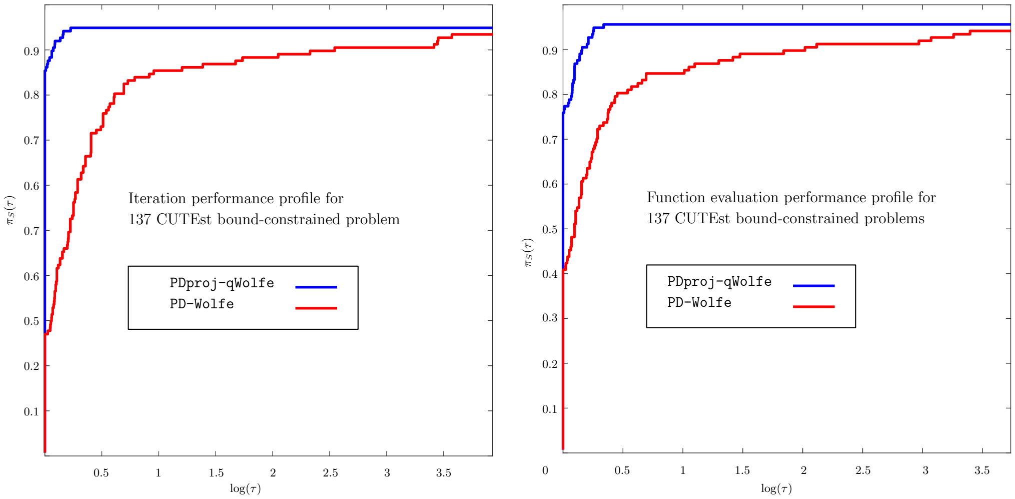

Figure 5.3: Performance profiles for two interior-point methods PD-Wolfe and PDproj-qWolfe. The figure gives the performance profiles for the total number of iterations and function evaluations required to solve 137 bound-constrained problems from the CUTEst test set.

5.3 The Projected-Search Interior Method for Nonlinear Optimization

5.3.1 The implementation

Numerical results were obtained for Matlab implementations of three variants of the shifted interior method. Algorithm pdb is an implementation of the shifted primal-dual method of Gill, Kungurtsev and Robinson [46]; pdbAll is the primal-dual method with shifts on both the primal and dual variables; and pdProj is the projected-search interior method proposed in Sections 4.3–4.5. Algorithms pdb and pdbAll are implemented with a flexible Armijo line search in which the step length is chosen to satisfy the conditions (4.16a)–(4.16d) with $\psi_k(\alpha\,;\mu)$ and $\phi_k(\alpha\,;\mu)$ given by $M\big(v_k+\alpha\mathit{\Delta} v_k\,;\mu\big)$ and $\|F\big(v_k+\alpha\mathit{\Delta} v_k\,;\mu\big)\|$. Exact second derivatives were used for all the runs.

The iterates were terminated at the first point that satisfied the conditions $e_P(x,s) < \tau_P$ and $e_D(x,s,y,w) < \tau_D$, where e_P and e_D are the primal and dual infeasibilities

$$e_P(x,s) = \left\| \begin{pmatrix} \min\{\,0,\ s\,\} \\ \|c(x)-s\|_\infty / \max\{\,1,\ \|s\|_\infty\,\} \end{pmatrix} \right\|_\infty, \tag{5.2a}$$

and

$$e_D(x,s,y,w) = \left\| \begin{pmatrix} \|\nabla f(x) - J(x)^{\mathrm{T}} y\|_\infty / \sigma \\ \|w - y\|_\infty \\ w \cdot \min\left\{ 1,\, s \right\} \end{pmatrix} \right\|_\infty , \tag{5.2b}$$

with $\sigma = \max\left\{ 1,\, \|\nabla f(x)\|,\, \max\left\{ 1,\, \|y\| \right\} \|J(x)\|_\infty \right\}$. Similarly, the iterates were terminated at an infeasible stationary point (x,s) if $e_P(x,s) > \tau_P$, $\min\left\{ 0,\, s \right\} \le \tau_P$ and $e_I(x,s) \le \tau_{\text{inf}}$, where

$$e_I(x,s) = \left\| J(x)^{\mathrm{T}} (c(x) - s) \cdot \min\left\{ 1,\, s \right\} \right\|_\infty / \sigma. \tag{5.3}$$

5.3.2 Numerical results

The results were obtained for optimization problems from the CUTEst test collection. The runs were done using Matlab version R2020a on an iMac Pro with a 3.0 GHz Intel Xeon W processor and 128 GB of 800 MHz DDR4 RAM running macOS, version 10.14.6 (64 bit). Results were obtained for five subsets of problems from the CUTEst test collection. The subsets consisted of 135 problems with a general nonlinear objective and upper and lower bounds on the variables (problems BC); 212 problems with a general nonlinear objective, general linear constraints and bounds on the variables (problems LC); 124 problems formulated by Hock and Schittkowski ([62]) (problems HS); 372 problems with a general nonlinear objective, general linear and nonlinear constraints and bounds on the variables (problems NC); and 117 problems with a quadratic objective, general linear constraints and bounds on the variables (problems QP). The BC, LC, NC and QP subsets were selected based on the number of variables and general constraints. In particular, a problem was chosen if the associated KKT system was of the order of 1000 or less. The same criterion was used to set the dimension of those problems for which the problem size can be specified. The nonsmooth problem HS87 was excluded from the Hock-Schittkowski problems. Exact second derivatives were used for all the runs.

Each CUTEst problem may be written in the form

$$\underset{x}{\text{minimize}} \;\; f(x) \quad \text{subject to} \quad \begin{pmatrix} \ell^X \\ \ell^S \end{pmatrix} \le \begin{pmatrix} x \\ c(x) \end{pmatrix} \le \begin{pmatrix} u^X \\ u^S \end{pmatrix}, \tag{5.4}$$

where $c : \mathbb{R}^n \mapsto \mathbb{R}^m$, $f : \mathbb{R}^n \mapsto \mathbb{R}$, and (ℓ^X, ℓ^S) and (u^X, u^S) are constant vectors of lower and upper bounds. In this format, a fixed variable or an equality constraint has the same value for its upper and lower bounds. A variable or constraint with no upper or lower limit is indicated by a bound of $\pm 10^{20}$.

The approximate Newton equations for problem (5.4) are derived in Appendix A. As is the case for problem (NIPs) the principal work at each iteration is the solution of a reduced $(n+m)\times(n+m)$ KKT system analogous to (4.14). Each KKT matrix was factored using the Matlab built-in command LDL. If this matrix was singular or had more than m negative eigenvalues, the Hessian of the Lagrangian H was modified using the method of Wächter and Biegler [84, Algorithm IC, p. 36], which factors the KKT matrix with δI_n added to H. At any given iteration the value of δ is increased from zero if necessary until the inertia of the KKT matrix is correct.

All three Matlab implementations were initialized with identical parameter values that were chosen based on the empirical performance on the entire collection of problems. A summary of the values is given in Table 5.1. The initial primal-dual estimate $(x_0,\, y_0)$ was based on the default initial values supplied by CUTEst. If necessary, x_0 was projected onto the set $\left\{\, x : \ell^X \le x \le u^X \,\right\}$ to ensure feasibility with respect to the bounds on x. The iterates were terminated at the first point that satisfied the conditions (5.2a)–(5.2b) or (5.3) defined in terms of the constraints associated with problem (5.4).

Table 5.1: Control parameters for Algorithms **pdb**, **pdbAll** and **pdProj**.

Parameter	Description	Value
$s_{\max}$, $y_{\max}$, $w_{\max}$	Maximum allowed y^E, w^E, s^E	1.0e+6
μ_0^P	Initial penalty parameter for Algorithm 5	1.0e-4
μ_0^L	Initial flexible line-search penalty parameter for Algorithm 5	1.0
μ_0^B	Initial barrier parameter for Algorithm 5	1.0e-4
τ_0	Initial termination tolerance for specifying an M-iterate	0.5
τ_P	Primal feasibility tolerance (5.2a)	1.0e-4
τ_D	Dual feasibility tolerance (5.2b)	1.0e-4
τ_{inf}	Infeasible stationary point tolerance (5.3)	1.0e-4
$\chi_0^{\max}$	Initial target for an O-iteration	1.0e+3
η_A	Line-search Armijo sufficient reduction	1.0e-2
η_F	Line-search sufficient reduction for $\|F\|$	1.0e-2
γ_A	Line-search factor for reducing an Armijo step	1.0e-3
f_{unb}	Unbounded objective	1.0e-9
$M_{\max}$	Constants in line-search tolerance (4.16a) and (4.16b)	1.0e+12
$F_{\max}$	Constant in the line-search tolerance (4.16c)	1.0e+8
σ	Bound perturbation in the definition of Ω_k (4.15)	0.8
$k_{\max}$	Iteration limit for Algorithm 5	500

Figures 5.4–5.8 present the performance profiles for the total number of iterations and function evaluations required to solve the 135 BC problems, 212 LC problems, 124 HS problems, 372 NC problems, and 117 QP problems successively. More details of the runs used to generate

the performance profiles are given by Gill and Zhang [55]. The profiles show that the projected-search interior method `pdProj` requires substantially fewer iterations and function evaluations than the other two methods `pdb` and `pdbAll`. As a result, the `pdProj` method also necessitates fewer computations of search directions. In particular, results from solving the 117 QP problems suggest that the `pdProj` method is especially well-suited to solving the quadratic programming subproblem in a sequential quadratic programming method for nonlinear optimization.

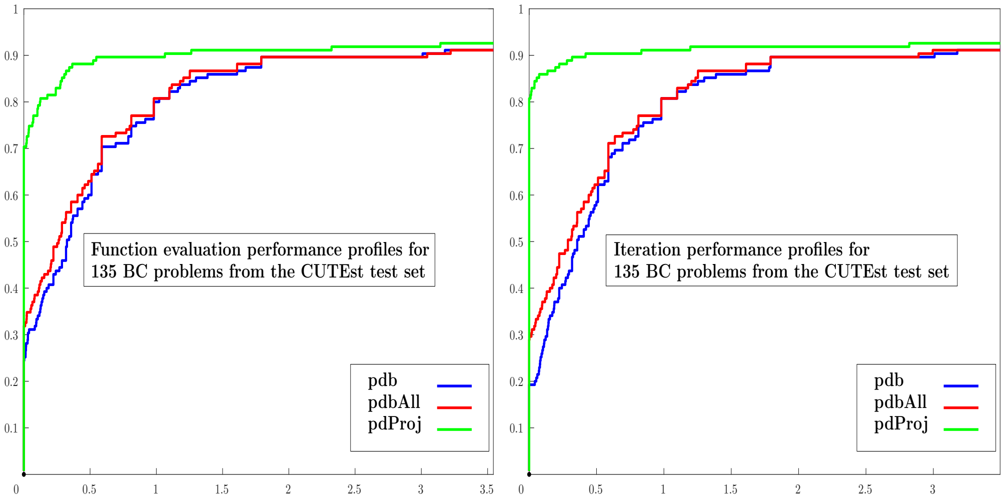

Figure 5.4: Performance profiles for the primal-dual interior algorithms `pdb`, `pdbAll` and `pdProj` applied to 135 bound-constrained (BC) problems from the CUTEst test set. The left figure gives the profiles for the number of function evaluations. The right figure gives the profiles for the number of iterations.

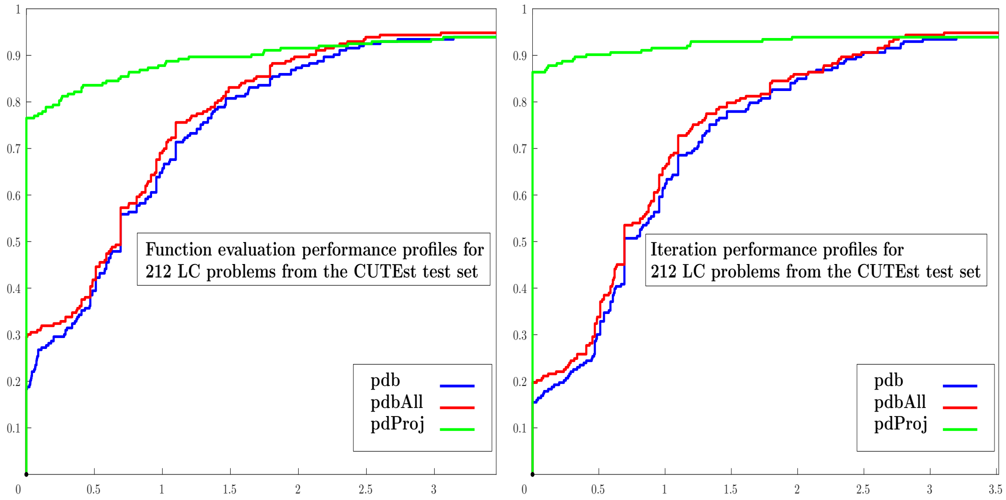

Figure 5.5: Performance profiles for the primal-dual interior algorithms `pdb`, `pdbAll` and `pdProj` applied to 212 linearly constrained (LC) problems from the CUTEst test set. The left figure gives the profiles for the number of function evaluations. The right figure gives the profiles for the number of iterations.

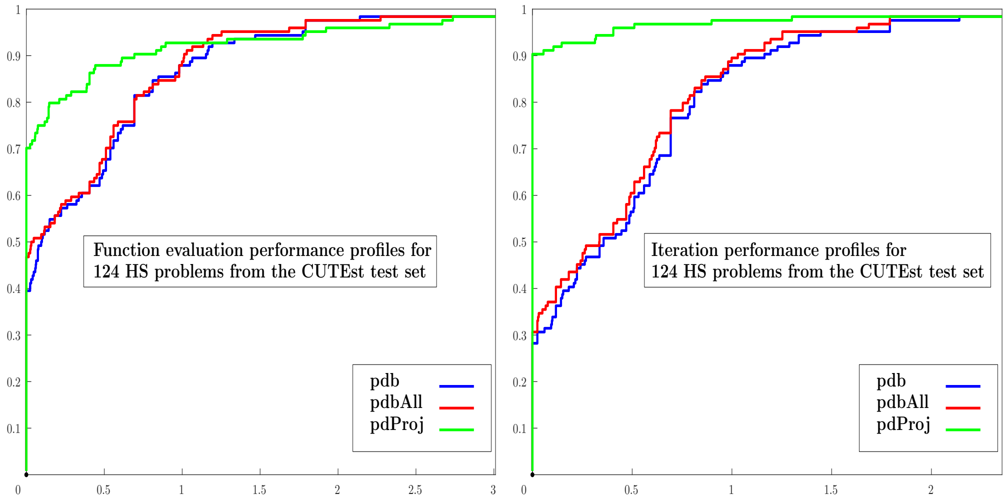

Figure 5.6: Performance profiles for the primal-dual interior algorithms `pdb`, `pdbAll` and `pdProj` applied to 124 Hock-Schittkowski (HS) problems from the CUTEst test set. The left figure gives the profiles for the number of function evaluations. The right figure gives the profiles for the number of iterations.

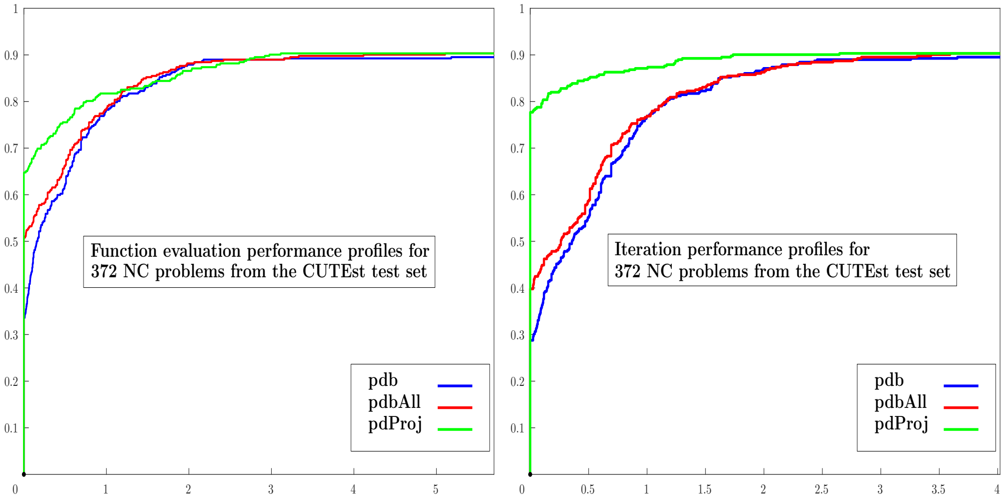

Figure 5.7: Performance profiles for the primal-dual interior algorithms `pdb`, `pdbAll` and `pdProj` applied to 372 nonlinearly constrained (NC) problems from the CUTEst test set. The left figure gives the profiles for the number of function evaluations. The right figure gives the profiles for the number of iterations.

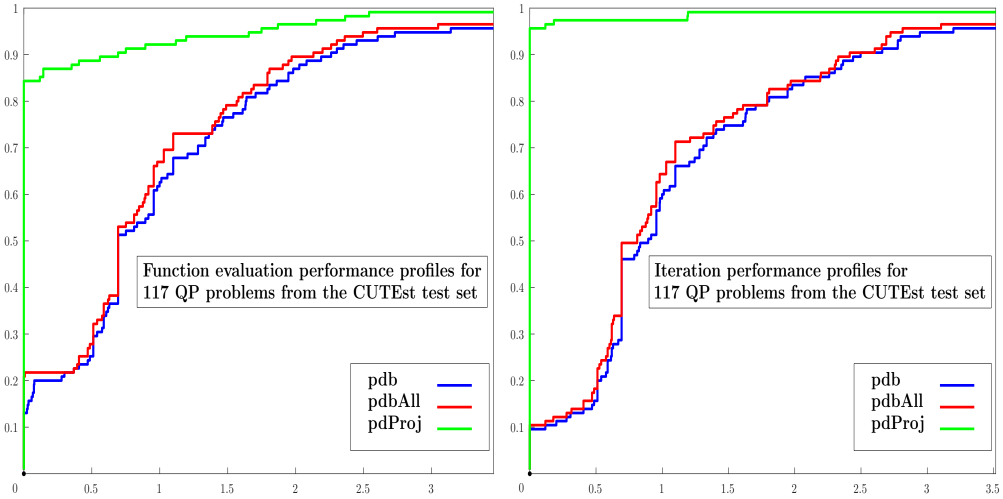

Figure 5.8: Performance profiles for the primal-dual interior algorithms `pdb`, `pdbAll` and `pdProj` applied to 117 quadratic programming (QP) problems from the CUTEst test set. The left figure gives the profiles for the number of function evaluations. The right figure gives the profiles for the number of iterations.

Appendix A

Equations of an All-Shifted Primal-Dual Penalty-Barrier Method for Nonlinear Optimization

A.1 Introduction

In Chapter 4, a primal-dual path-following method was proposed for nonlinearly constrained optimization, combining a shifted primal-dual path-following method with a projected-search method for bound-constrained optimization. The method involves the computation of an approximate Newton direction for a primal-dual penalty-barrier function that incorporates shifts on both the primal and dual variables. The focus of this appendix is the derivation of the primal-dual equations for this method when applied to general nonlinearly constrained problems. In particular, the equations are formulated for problems written in the general form:

$$\underset{x\in\mathbb{R}^n, s\in\mathbb{R}^m}{\text{minimize}} \quad f(x) \quad \text{subject to} \quad \begin{cases} c(x) - s = 0, & L_X s = h_X, \quad \ell^S \le L_L s, \quad L_U s \le u^S, \\ Ax - b = 0, & E_X x = b_X, \quad \ell^X \le E_L x, \quad E_U x \le u^X, \end{cases} \tag{NLP}$$

where A denotes a constant $m_A \times n$ matrix, and b, h_X, b_X, ℓ^S, u^S, ℓ^X and u^X are fixed vectors of dimension m_A, m_X, n_X, m_L, m_U, n_L and n_U, respectively. Similarly, L_X, L_L and L_U denote fixed matrices of dimension $m_X \times m$, $m_L \times m$ and $m_U \times m$, respectively, and E_X, E_L and E_U are fixed matrices of dimension $n_X \times n$, $n_L \times n$ and $n_U \times n$, respectively. Throughout the discussion, the functions $c : \mathbb{R}^n \mapsto \mathbb{R}^m$ and $f : \mathbb{R}^n \mapsto \mathbb{R}$ are assumed to be twice-continuously differentiable. The components of s may be interpreted as slack variables associated with the nonlinear constraints.

The quantity E_X denotes an $n_X \times n$ matrix formed from n_X independent rows of I_n, the identity matrix of order n. This implies that the equality constraints $E_X x = b_X$ fix n_X components of x at the corresponding values of b_X. Similarly, E_L and E_U denote $n_L \times n$ and $n_U \times n$ matrices formed from subsets of rows of I_n such that $E_X^T E_L = 0$, $E_X^T E_U = 0$, i.e., a variable is either fixed or free to move, possibly bounded by an upper or lower bound. Note that an x_j may be an unrestricted variable in the sense that it is neither fixed nor subject to an upper or lower bound, in which case e_j^T is not a row of E_X, E_L or E_U. Analogous definitions hold for L_X, L_L and L_U as subsets of rows of I_m. However, we impose the restriction that a given s_j must be either fixed or restricted by an upper or lower bound, i.e., there are no unrestricted slacks[1]. Let E_F denote the matrix of rows of I_n that are not rows of E_X, and let L_F denote the matrix of rows of I_m that are not rows of L_X. If $n_F = n - n_X$ and $m_F = m - m_X$, then E_F and L_F are $n_F \times n$ and $m_F \times m$ respectively. Note that $n_L + n_U$ may be less than n_F, but m_F must equal $m_L + m_U$. The matrices $\begin{pmatrix} E_X^T & E_F^T \end{pmatrix}$ and $\begin{pmatrix} L_X^T & L_F^T \end{pmatrix}$ are column permutations of I_n and I_m. Moreover, there are $n \times n$ and $m \times m$ permutation matrices P_x and P_s such that

$$P_x = \begin{pmatrix} E_F \\ E_X \end{pmatrix} \quad \text{and} \quad P_s = \begin{pmatrix} L_F \\ L_X \end{pmatrix},$$

with $E_F E_F^T = I_F^x$, $E_X E_X^T = I_X^x$, and $E_F E_X^T = 0$, and $L_F L_F^T = I_F^s$, $L_X L_X^T = I_X^s$, and $L_F L_X^T = 0$.

[1] This is not a significant restriction because a "free" slack is equivalent to a unrestricted nonlinear constraint, which may be discarded from the problem. The shifted primal-dual penalty-barrier equations can be derived without this restriction, but the derivation is beyond the scope of this work.

All general inequality constraints are imposed indirectly using a shifted primal-dual barrier function. The general equality constraints $c(x) - s = 0$ and $Ax = b$ are enforced using an primal-dual augmented Lagrangian algorithm, which implies that the equalities are satisfied in the limit. The exception to this is when the constraints $E_X x = b_X$, and $L_X s = h_X$ are used to fix a subset of the variables and slacks. These bounds are enforced at every iterate.

An equality constraint $c_i(x) = 0$ may be handled by introducing the slack variable s_i and writing the constraint as the two constraints $c_i(x) - s_i = 0$ and $s_i = 0$. In this case the ith coordinate vector e_i can be included as a row of L_X. Linear *inequality* constraints must be included as part of c. A linear equality constraint can be either included with the nonlinear equality constraints or the matrix A. The constraints involving A may be used to temporarily fix a subset of the variables at their bounds without altering the underlying structure of the approximate Newton equations. In this case, the associated rows of A are rows of the identity matrix.

The optimality conditions for problem (NLP) are given in Section A.2. The shifted path-following equations are formulated in Section A.3. The shifted primal-dual penalty-barrier function associated with problem is discussed in Section A.4. This function serves as a merit function for the projected-search method. The equations are formulated in Sections A.5 and A.6, and summarized in Section A.7.

A.2 Optimality Conditions

The first-order KKT conditions for problem (NLP) are

$$\left.\begin{array}{rrr}
\nabla f(x^*) - J(x^*)^{\mathrm{T}} y^* - A^{\mathrm{T}} v^* - E_X^{\mathrm{T}} z_X^* - E_L^{\mathrm{T}} z_1^* + E_U^{\mathrm{T}} z_2^* = 0, & z_1^* \geq 0, & z_2^* \geq 0, \\
y^* - L_X^{\mathrm{T}} w_X^* - L_L^{\mathrm{T}} w_1^* + L_U^{\mathrm{T}} w_2^* = 0, & w_1^* \geq 0, & w_2^* \geq 0, \\
c(x^*) - s^* = 0, & & L_X s^* - h_X = 0, \\
 & Ax^* - b = 0, & E_X x^* - b_X = 0, \\
E_L x^* - \ell^X \geq 0, & u^X - E_U x^* \geq 0, & \\
L_L s^* - \ell^S \geq 0, & u^S - L_U s^* \geq 0, & \\
z_1^* \cdot (E_L x^* - \ell^X) = 0, & z_2^* \cdot (u^X - E_U x^*) = 0, & \\
w_1^* \cdot (L_L s^* - \ell^S) = 0, & w_2^* \cdot (u^S - L_U s^*) = 0, &
\end{array}\right\} \tag{A.1}$$

where y^*, w_X^*, and z_X^* are the multipliers for the equality constraints $c(x) - s = 0$, $L_X s^* = h_X$ and $E_X x^* = b_X$, and z_1^*, z_2^*, w_1^* and w_2^* may be interpreted as the Lagrange multipliers for the inequality constraints $E_L x - \ell^X \geq 0$, $u^X - E_U x \geq 0$, $L_L s - \ell^S \geq 0$ and $u^S - L_U s \geq 0$, respectively. The components of v^* are the multipliers for the linear equality constraints $Ax = b$.

The discussion that follows makes extensive use of the auxiliary quantities

$$x_1 = E_L x - \ell^X, \quad x_2 = u^X - E_U x, \quad s_1 = L_L s - \ell^S, \quad \text{and} \quad s_2 = u^S - L_U s. \tag{A.2}$$

In some cases x_1, x_2, s_1 and s_2 are used to simplify the expressions appearing in certain equations, in others they are regarded as independent variables associated with the problem

$$\left.\begin{array}{lllll} \underset{x,x_1,x_2,s,s_1,s_2}{\text{minimize}} & f(x) & & & \\ \text{subject to} & c(x) - s = 0, & Ax - b = 0, & & \\ & E_L x - x_1 = \ell^X, & L_L s - s_1 = \ell^S, & x_1 \ge 0, & s_1 \ge 0, \\ & E_U x + x_2 = u^X, & L_U s + s_2 = u^S, & x_2 \ge 0, & s_2 \ge 0, \\ & E_X x - b_X = 0, & L_X s - h_X = 0, & & \end{array}\right\} \qquad \text{(NLP')}$$

which is equivalent to problem (NLP). In this case, the dual variables z_1^*, z_2^*, w_1^*, and w_2^* associated with the optimality conditions (A.1) are the Lagrange multipliers for the inequality constraints $x_1 \ge 0$, $x_2 \ge 0$, $s_1 \ge 0$, and $s_2 \ge 0$, respectively.

In the derivations that follow, the vectors z and w are defined as

$$z = E_X^{\mathrm{T}} z_X + E_L^{\mathrm{T}} z_1 - E_U^{\mathrm{T}} z_2, \quad \text{and} \quad w = L_X^{\mathrm{T}} w_X + L_L^{\mathrm{T}} w_1 - L_U^{\mathrm{T}} w_2. \tag{A.3}$$

A.3 The Path-Following Equations

Penalty and barrier methods are closely related to path-following methods. These methods approximate a continuous path that passes through a solution of (NLP). In the simplest case, the path is parameterized by a positive scalar parameter that may be interpreted as a perturbation for the optimality conditions for the problem (NLP).

Let z_1^E and z_2^E, w_1^E and w_2^E denote nonnegative estimates of z_1^* and z_2^*, w_1^* and w_2^*. Similarly, let v^E, x^E and s^E denote estimates of v^*, x^* and s^*. Given small positive scalars μ^P, μ^A and μ^B, consider the perturbed optimality conditions

$$\left.\begin{aligned}
\nabla f(x) - J(x)^{\mathrm{T}} y - A^{\mathrm{T}} v - E_X^{\mathrm{T}} z_X - E_L^{\mathrm{T}} z_1 + E_U^{\mathrm{T}} z_2 &= 0, & z_1 \ge 0, &\quad z_2 \ge 0,\\
y - L_X^{\mathrm{T}} w_X - L_L^{\mathrm{T}} w_1 + L_U^{\mathrm{T}} w_2 &= 0, & w_1 \ge 0, &\quad w_2 \ge 0,\\
c(x) - s &= \mu^P (y^E - y), & E_X x - b_X = 0, &\quad L_X s - h_X = 0,\\
Ax - b &= \mu^A (v^E - v), & &\\
E_L x - \ell^X &\ge 0, & u^X - E_U x \ge 0, &\\
L_L s - \ell^S &\ge 0, & u^S - L_U s \ge 0, &\\
z_1 \cdot (E_L x - \ell^X) &= \mu^B (z_1^E - z_1) + \mu^B (E_L x^E - E_L x), & &\\
z_2 \cdot (u^X - E_U x) &= \mu^B (z_2^E - z_2) + \mu^B (E_U x - E_U x^E), & &\\
w_1 \cdot (L_L s - \ell^S) &= \mu^B (w_1^E - w_1) + \mu^B (L_L s^E - L_L s), & &\\
w_2 \cdot (u^S - L_U s) &= \mu^B (w_2^E - w_2) + \mu^B (L_U s - L_U s^E). & &
\end{aligned}\right\} \tag{A.4}$$

Let v_P denote the vector of variables $v_P = (x,\, s,\, y,\, v,\, w_X,\, z_X,\, z_1,\, z_2,\, w_1,\, w_2)$. The primal-dual path-following equations are given by $F(v_P) = 0$, with

$$F(v_P) = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y - A^{\mathrm{T}} v - E_X^{\mathrm{T}} z_X - E_L^{\mathrm{T}} z_1 + E_U^{\mathrm{T}} z_2 \\ y - L_X^{\mathrm{T}} w_X - L_L^{\mathrm{T}} w_1 + L_U^{\mathrm{T}} w_2 \\ c(x) - s + \mu^P (y - y^E) \\ Ax - b + \mu^A (v - v^E) \\ E_X x - b_X \\ L_X s - h_X \\ z_1 \cdot (E_L x - \ell^X) + \mu^B (z_1 - z_1^E) + \mu^B (E_L x - E_L x^E) \\ z_2 \cdot (u^X - E_U x) + \mu^B (z_2 - z_2^E) + \mu^B (E_U x^E - E_U x) \\ w_1 \cdot (L_L s - \ell^S) + \mu^B (w_1 - w_1^E) + \mu^B (L_L s - L_L s^E) \\ w_2 \cdot (u^S - L_U s) + \mu^B (w_2 - w_2^E) + \mu^B (L_U s^E - L_U s) \end{pmatrix} = \begin{pmatrix} \nabla f(x) - J(x)^{\mathrm{T}} y - A^{\mathrm{T}} v - z \\ y - w \\ c(x) - s + \mu^P (y - y^E) \\ Ax - b + \mu^A (v - v^E) \\ E_X x - b_X \\ L_X s - h_X \\ z_1 \cdot (E_L x - \ell^X) + \mu^B (z_1 - z_1^E) + \mu^B (E_L x - E_L x^E) \\ z_2 \cdot (u^X - E_U x) + \mu^B (z_2 - z_2^E) + \mu^B (E_U x^E - E_U x) \\ w_1 \cdot (L_L s - \ell^S) + \mu^B (w_1 - w_1^E) + \mu^B (L_L s - L_L s^E) \\ w_2 \cdot (u^S - L_U s) + \mu^B (w_2 - w_2^E) + \mu^B (L_U s^E - L_U s) \end{pmatrix}, \tag{A.5}$$

where the first $n{+}m$ equations are written in terms of z and w such that $z = E_X^{\mathrm{T}} z_X + E_L^{\mathrm{T}} z_1 - E_U^{\mathrm{T}} z_2$ and $w = L_X^{\mathrm{T}} w_X + L_L^{\mathrm{T}} w_1 - L_U^{\mathrm{T}} w_2$. (To simplify the notation, the dependence of F on the parameters μ^A, μ^P, μ^B, x^E, s^E, y^E, v^E, z_1^E, z_2^E, w_1^E, w_2^E is omitted.) Any zero $(x,\, s,\, y,\, v,\, w_X,\, z_X,\, z_1,\, z_2,\, w_1,\, w_2)$ of F such that $\ell^X < E_L x$, $E_U x < u^X$, $\ell^S < L_L s$, $L_U < u^S$, $z_1 > 0$, $z_2 > 0$, $w_1 > 0$, and $w_2 > 0$ approximates a point satisfying the optimality conditions (A.1), with the approximation becoming increasingly accurate as the terms $\mu^P(y - y^E)$, $\mu^A(v - v^E)$, $\mu^B(E_L x^E - E_L x)$, $\mu^B(E_U x^E - E_U x)$, $\mu^B(L_L s^E - L_L s)$, $\mu^B(L_U s - L_L^2)$, $\mu^B(z_1 - z_1^E)$, $\mu^B(z_2 - z_2^E)$, $\mu^B(w_1 - w_1^E)$ and $\mu^B(w_2 - w_2^E)$ approach zero. For any sequence of x^E, s^E, z_1^E, z_2^E, w_1^E, w_2^E, v^E and y^E such that $x^E \to x^*$, $s^E \to s^*$, $z_1^E \to z_1^*$, $z_2^E \to z_2^*$, $w_1^E \to w_1^*$, $w_2^E \to w_2^*$, $v^E \to v^*$ and $y^E \to y^*$, it must hold that solutions $(x,\, s,\, y,\, v,\, z_1,\, z_2,\, w_1,\, w_2)$ of (A.4) must satisfy $z_1 \cdot (x - \ell^X) \to 0$, $z_2 \cdot (u^X - x) \to 0$, $w_1 \cdot (s - \ell^S) \to 0$, and $w_2 \cdot (u^S - s) \to 0$, This implies that any solution $(x,\, s,\, y,\, v,\, w_X,\, z_X,\, z_1,\, z_2,\, w_1,\, w_2)$ of (A.4) will approximate a solution of (A.1) independently of the values of μ^P, μ^A and μ^B (i.e., it is not necessary that $\mu^P \to 0$, $\mu^A \to 0$ and $\mu^B \to 0$).

If $v_P = (x,\, s,\, y,\, v,\, w_X,\, z_X,\, z_1,\, z_2,\, w_1,\, w_2)$ is a given approximate zero of $F(v_P)$ such that $\ell^X - \mu^B < E_L x$, $E_U x < u^X + \mu^B$, $\ell^S - \mu^B < L_L s$, $L_U s < u^S + \mu^B$, $z_1 > 0$, $z_2 > 0$, $w_1 > 0$, and $w_2 > 0$, the Newton equations for the change in variables $\Delta v_P = (\Delta x,$

Δs, Δy, Δv, Δw_X, Δz_X, Δz_1, Δz_2, Δw_1, Δw_2) are given by $F'(v_P)\Delta v_P = -F(v_P)$, with

$$F'(v_P) = \begin{pmatrix} H(x,y) & 0 & -J^{\mathrm{T}} & -A^{\mathrm{T}} & 0 & -E_X^{\mathrm{T}} & -E_L^{\mathrm{T}} & E_U^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & I_m & 0 & -L_X^{\mathrm{T}} & 0 & 0 & 0 & -L_L^{\mathrm{T}} & L_U^{\mathrm{T}} \\ J & -I_m & D_Y & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & 0 & 0 & D_A & 0 & 0 & 0 & 0 & 0 & 0 \\ E_X & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_X & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Z_1^{\mu} E_L & 0 & 0 & 0 & 0 & 0 & X_1^{\mu} & 0 & 0 & 0 \\ -Z_2^{\mu} E_U & 0 & 0 & 0 & 0 & 0 & 0 & X_2^{\mu} & 0 & 0 \\ 0 & W_1^{\mu} L_L & 0 & 0 & 0 & 0 & 0 & 0 & S_1^{\mu} & 0 \\ 0 & -W_2^{\mu} L_U & 0 & 0 & 0 & 0 & 0 & 0 & 0 & S_2^{\mu} \end{pmatrix}, \tag{A.6}$$

where

$$\left.\begin{array}{llll} X_1^{\mu} = \operatorname{diag}(x_1 + \mu^B e), & X_2^{\mu} = \operatorname{diag}(x_2 + \mu^B e), & S_1^{\mu} = \operatorname{diag}(s_1 + \mu^B e), & S_2^{\mu} = \operatorname{diag}(s_2 + \mu^B e), \\ Z_1^{\mu} = \operatorname{diag}(z_1 + \mu^B e), & Z_2^{\mu} = \operatorname{diag}(z_2 + \mu^B e), & W_1^{\mu} = \operatorname{diag}(w_1 + \mu^B e), & W_2^{\mu} = \operatorname{diag}(w_2 + \mu^B e), \end{array}\right\} \tag{A.7}$$

with x_1, x_2, s_1 and s_2 given by (A.2). Any s may be written as $s = L_F^{\mathrm{T}} s_F + L_X^{\mathrm{T}} s_X$, where L_F are the rows of I_m orthogonal to the rows of L_X, i.e., $L_F^{\mathrm{T}} L_X = 0$. The vectors s_F and s_X are the components of s corresponding to the "free" and "fixed" components of s, respectively. The variables $L_L s$ and $L_U s$ form a subset of s_F. Throughout, we assume that s satisfies $L_X s - h_X = 0$, in which case $\Delta s_X = 0$ and Δs satisfies

$$\Delta s = L_F^{\mathrm{T}} \Delta s_F + L_X^{\mathrm{T}} \Delta s_X = L_F^{\mathrm{T}} \Delta s_F.$$

Similarly, any x may be written as $x = E_F^{\mathrm{T}} x_F + E_X^{\mathrm{T}} x_X$, where x_F and x_X denote the components of x corresponding to the "free" and "fixed variables", respectively. The variables $E_L x$ and $E_U x$ form a subset of x_F. Throughout, we assume that x_X satisfies

$E_X x - b_X = 0$, in which case $\varDelta x_X = 0$ and $\varDelta x$ satisfies

$$\varDelta x = E_F^{\mathrm{T}} \varDelta x_F + E_X^{\mathrm{T}} \varDelta x_X = E_F^{\mathrm{T}} \varDelta x_F.$$

After premultiplying the first and fifth blocks of equations of (A.6) by E_F and L_F respectively, and substituting $\varDelta x = E_F^{\mathrm{T}} \varDelta x_F$ and $\varDelta s = L_F^{\mathrm{T}} \varDelta s_F$, the equations (A.6) can be written in the reduced form $\widehat{F}'(v_F)\varDelta v_F = -\widehat{F}(v_F)$, where $\varDelta v_F = (\varDelta x_F,\ \varDelta s_F,\ \varDelta y,\ \varDelta v,\ \varDelta z_1,\ \varDelta z_2,\ \varDelta w_1,\ \varDelta w_2)$,

$$\begin{pmatrix} H_F & 0 & -J_F^{\mathrm{T}} & -A_F^{\mathrm{T}} & -E_{LF}^{\mathrm{T}} & E_{UF}^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & L_F & 0 & 0 & 0 & -L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_F & -L_F^{\mathrm{T}} & D_Y & 0 & 0 & 0 & 0 & 0 \\ A_F & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\ Z_1^\mu E_{LF} & 0 & 0 & 0 & X_1^\mu & 0 & 0 & 0 \\ -Z_2^\mu E_{UF} & 0 & 0 & 0 & 0 & X_2^\mu & 0 & 0 \\ 0 & W_1^\mu L_{LF} & 0 & 0 & 0 & 0 & S_1^\mu & 0 \\ 0 & -W_2^\mu L_{UF} & 0 & 0 & 0 & 0 & 0 & S_2^\mu \end{pmatrix} \begin{pmatrix} \varDelta x_F \\ \varDelta s_F \\ \varDelta y \\ \varDelta v \\ \varDelta z_1 \\ \varDelta z_2 \\ \varDelta w_1 \\ \varDelta w_2 \end{pmatrix}$$

$$= - \begin{pmatrix} g_F - J_F^{\mathrm{T}} y - A_F^{\mathrm{T}} v - E_{LF}^{\mathrm{T}} z_1 + E_{UF}^{\mathrm{T}} z_2 \\ y_F - L_{LF}^{\mathrm{T}} w_1 + L_{UF}^{\mathrm{T}} w_2 \\ c(x) - s + \mu^P (y - y^E) \\ Ax - b + \mu^A (v - v^E) \\ z_1 \cdot (E_L x - \ell^X) + \mu^B (z_1 - z_1^E) + \mu^B (E_L x - E_L x^E) \\ z_2 \cdot (u^X - E_U x) + \mu^B (z_2 - z_2^E) + \mu^B (E_U x^E - E_U x) \\ w_1 \cdot (L_L s - \ell^S) + \mu^B (w_1 - w_1^E) + \mu^B (L_L s - L_L s^E) \\ w_2 \cdot (u^S - L_U s) + \mu^B (w_2 - w_2^E) + \mu^B (L_U s^E - L_U s) \end{pmatrix},$$

where $H_F = E_F H E_F^{\mathrm{T}}$, $J_F = J(x)E_F^{\mathrm{T}}$, $A_F = AE_F^{\mathrm{T}}$, $g_F = E_F \nabla f(x)$, $E_{LF} = E_L E_F^{\mathrm{T}}$, $E_{UF} = E_U E_F^{\mathrm{T}}$, $y_F = L_F y$, $L_{LF} = L_L L_F^{\mathrm{T}}$ and $L_{UF} = L_U L_F^{\mathrm{T}}$. The matrices J_F, A_F, E_{LF} and E_{UF} are the columns of $J(x)$, A, E_L and E_U associated with the "free" components of x. The matrices L_{LF} and L_{UF} are the columns of L_L and L_U associated with the "free" components of s. Then scaling the last four blocks of equations by (respectively) $(Z_1^{\mu})^{-1}$, $(Z_2^{\mu})^{-1}$, $(W_1^{\mu})^{-1}$ and $(W_2^{\mu})^{-1}$ gives

$$\begin{pmatrix} H_F & 0 & -J_F^{\mathrm{T}} & -A_F^{\mathrm{T}} & -E_{LF}^{\mathrm{T}} & E_{UF}^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & L_F & 0 & 0 & 0 & -L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_F & -L_F^{\mathrm{T}} & D_Y & 0 & 0 & 0 & 0 & 0 \\ A_F & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & D_1^Z & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & D_2^Z & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & D_1^W & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & D_2^W \end{pmatrix} \begin{pmatrix} \Delta x_F \\ \Delta s_F \\ \Delta y \\ \Delta v \\ \Delta z_1 \\ \Delta z_2 \\ \Delta w_1 \\ \Delta w_2 \end{pmatrix} = - \begin{pmatrix} g_F - J_F^{\mathrm{T}} y - A_F^{\mathrm{T}} v - E_{LF}^{\mathrm{T}} z_1 + E_{UF}^{\mathrm{T}} z_2 \\ y_F - L_{LF}^{\mathrm{T}} w_1 + L_{UF}^{\mathrm{T}} w_2 \\ c(x) - s + \mu^P (y - y^E) \\ Ax - b + \mu^A (v - v^E) \\ D_1^Z (z_1 - \pi_1^Z) \\ D_2^Z (z_2 - \pi_2^Z) \\ D_1^W (w_1 - \pi_1^W) \\ D_2^W (w_2 - \pi_2^W) \end{pmatrix}, \qquad \text{(A.8)}$$

where $A_F = AE_F^{\mathrm{T}}$ are the columns of A associated with the "free" components of x, and

$$\begin{aligned} D_Y &= \mu^P I_m, & \pi^Y &= y^E - \frac{1}{\mu^P}(c - s), & D_A &= \mu^A I_A, & \pi^V &= v^E - \frac{1}{\mu^A}(Ax - b), \\ D_1^W &= S_1^{\mu}(W_1^{\mu})^{-1}, & \pi_1^W &= \mu^B (S_1^{\mu})^{-1}(w_1^E - s_1 + s_1^E), & D_1^Z &= X_1^{\mu}(Z_1^{\mu})^{-1}, & \pi_1^Z &= \mu^B (X_1^{\mu})^{-1}(z_1^E - x_1 + x_1^E), \\ D_2^W &= S_2^{\mu}(W_2^{\mu})^{-1}, & \pi_2^W &= \mu^B (S_2^{\mu})^{-1}(w_2^E - s_2 + s_2^E), & D_2^Z &= X_2^{\mu}(Z_2^{\mu})^{-1}, & \pi_2^Z &= \mu^B (X_2^{\mu})^{-1}(z_2^E - x_2 + x_2^E), \end{aligned}$$

with auxiliary quantities

$$x_1^E = E_L x^E - \ell^X, \quad x_2^E = u^X - E_U x^E, \quad s_1^E = L_L s^E - \ell^S, \quad \text{and} \quad s_2^E = u^S - L_U s^E.$$

Given the definitions (A.3), the vectors Δs and Δw_X are recovered as $\Delta s = L_F^{\mathrm{T}} \Delta s_F$ and $\Delta w_X = [\, y + \Delta y - w \,]_X$. Similarly, Δx and Δz_X are recovered as $\Delta x = L_F^{\mathrm{T}} \Delta x_F$ and $\Delta z_X = [\, g + H\Delta x - J^{\mathrm{T}}(y + \Delta y) - z \,]_X$.

A.4 A Shifted Primal-Dual Penalty-Barrier Function

Consider the shifted primal-dual penalty-barrier problem applied to (NP):

$$
\begin{array}{lllllll}
\underset{\substack{x,x_1,x_2,s,s_1,s_2,\\ y,v,z_1,z_2,w_1,w_2}}{\text{minimize}} & \multicolumn{6}{l}{M(x,x_1,x_2,s,s_1,s_2,y,v,w_1,w_2\,;\mu^P,\mu^B,y^E,v^E,w_1^E,w_2^E)} \\
\text{subject to} & E_Lx - x_1 = \ell^X, \quad L_Ls - s_1 = \ell^S, & x_1 + \mu^B e > 0, & z_1 + \mu^B e > 0, & s_1 + \mu^B e > 0, & w_1 + \mu^B e > 0, \\
 & E_Ux + x_2 = u^X, \quad L_Us + s_2 = u^S, & x_2 + \mu^B e > 0, & z_2 + \mu^B e > 0, & s_2 + \mu^B e > 0, & w_2 + \mu^B e > 0, \\
 & E_Xx - b_X = 0, \quad L_Xs - h_X = 0, & & & &
\end{array}
$$

where $M(x,x_1,x_2,s,s_1,s_2,y,v,z_1,z_2,w_1,w_2\,;\mu^P,\mu^B,y^E,v^E,z_1^E,z_2^E,w_1^E,w_2^E)$ is the shifted primal-dual penalty-barrier function

$$
\begin{aligned}
f(x) - \big(c(x)-s\big)^{\mathrm{T}} y^E + \frac{1}{2\mu^P}\|c(x)-s\|^2 + \frac{1}{2\mu^P}\|c(x)-s+\mu^P(y-y^E)\|^2 \\
- (Ax-b)^{\mathrm{T}} v^E + \frac{1}{2\mu^A}\|Ax-b\|^2 + \frac{1}{2\mu^A}\|Ax-b+\mu^A(v-v^E)\|^2 \\
- \sum_{j=1}^{n_L}\Big\{\mu^B([\,z_1^E\,]_j + [\,x_1^E\,]_j + \mu^B)\ln\big([\,z_1+\mu^B e\,]_j[\,x_1+\mu^B e\,]_j^2\big) - [\,z_1\cdot(x_1+\mu^B e)\,]_j - 2\mu^B[\,x_1\,]_j\Big\} \\
- \sum_{j=1}^{n_U}\Big\{\mu^B([\,z_2^E\,]_j + [\,x_2^E\,]_j + \mu^B)\ln\big([\,z_2+\mu^B e\,]_j[\,x_2+\mu^B e\,]_j^2\big) - [\,z_2\cdot(x_2+\mu^B e)\,]_j - 2\mu^B[\,x_2\,]_j\Big\} \\
- \sum_{i=1}^{m_L}\Big\{\mu^B([\,w_1^E\,]_i + [\,s_1^E\,]_i + \mu^B)\ln\big([\,w_1+\mu^B\,]_i[\,s_1+\mu^B e\,]_i^2\big) - [\,w_1\cdot(s_1+\mu^B e)\,]_i - 2\mu^B[\,s_1\,]_i\Big\} \\
- \sum_{i=1}^{m_U}\Big\{\mu^B([\,w_2^E\,]_i + [\,s_2^E\,] + \mu^B)\ln\big([\,w_2+\mu^B\,]_i[\,s_2+\mu^B e\,]_i^2\big) - [\,w_2\cdot(s_2+\mu^B e)\,]_i - 2\mu^B[\,s_2\,]_i\Big\}. \qquad \text{(A.9)}
\end{aligned}
$$

The gradient may be written as

$$\nabla M(x, x_1, x_2, s, s_1, s_2, y, v, z_1, z_2, w_1, w_2) = \begin{pmatrix} \nabla f(x) - A^{\mathrm{T}}\big(2(v^{E} - \frac{1}{\mu^{A}}(Ax - b)) - v\big) - J(x)^{\mathrm{T}}\big(2(y^{E} - \frac{1}{\mu^{P}}(c - s)) - y\big) \\ z_1 + 2\mu^{B} e - 2\mu^{B}(X_1^{\mu})^{-1}(z_1^{E} + x_1^{E} + \mu^{B} e) \\ z_2 + 2\mu^{B} e - 2\mu^{B}(X_2^{\mu})^{-1}(z_2^{E} + x_2^{E} + \mu^{B} e) \\ 2\big(y^{E} - \frac{1}{\mu^{P}}(c - s)\big) - y \\ w_1 + 2\mu^{B} e - 2\mu^{B}(S_1^{\mu})^{-1}(w_1^{E} + s_1^{E} + \mu^{B} e) \\ w_2 + 2\mu^{B} e - 2\mu^{B}(S_2^{\mu})^{-1}(w_2^{E} + s_2^{E} + \mu^{B} e) \\ c(x) - s + \mu^{P}(y - y^{E}) \\ Ax - b + \mu^{A}(v - v^{E}) \\ x_1 + \mu^{B} e - \mu^{B}(Z_1^{\mu})^{-1}(z_1^{E} + x_1^{E} + \mu^{B} e) \\ x_2 + \mu^{B} e - \mu^{B}(Z_2^{\mu})^{-1}(z_2^{E} + x_2^{E} + \mu^{B} e) \\ s_1 + \mu^{B} e - \mu^{B}(W_1^{\mu})^{-1}(w_1^{E} + s_1^{E} + \mu^{B} e) \\ s_2 + \mu^{B} e - \mu^{B}(W_2^{\mu})^{-1}(w_2^{E} + s_2^{E} + \mu^{B} e) \end{pmatrix},$$

where X_1^μ, X_2^μ, S_1^μ, S_2^μ, Z_1^μ, Z_2^μ, W_1^μ and W_2^μ are defined in (A.7). Equivalently,

$$\nabla M = \begin{pmatrix} \nabla f(x) - A^{\mathrm{T}}\big(\pi^{V} + (\pi^{V} - v)\big) - J(x)^{\mathrm{T}}\big(\pi^{Y} + (\pi^{Y} - y)\big) \\ z_1 - 2\pi_1^{Z} \\ z_2 - 2\pi_2^{Z} \\ \pi^{Y} + (\pi^{Y} - y) \\ w_1 - 2\pi_1^{W} \\ w_2 - 2\pi_2^{W} \\ -D_{Y}(\pi^{Y} - y) \\ -D_{A}(\pi^{V} - v) \\ -D_1^{Z}(\pi_1^{Z} - z_1) \\ -D_2^{Z}(\pi_2^{Z} - z_2) \\ -D_1^{W}(\pi_1^{W} - w_1) \\ -D_2^{W}(\pi_2^{W} - w_2) \end{pmatrix}.$$

The Hessian $\nabla^2 M(x, x_1, x_2, s, s_1, s_2, y, v, z_1, z_2, w_1, w_2)$ is given by

$$\begin{pmatrix}
H_1 & 0 & 0 & -2J^{\mathrm{T}}D_{\scriptscriptstyle Y}^{-1} & 0 & 0 & J^{\mathrm{T}} & A^{\mathrm{T}} & 0 & 0 & 0 & 0 \\
0 & 2G_1^{\scriptscriptstyle X} & 0 & 0 & 0 & 0 & -I_m & 0 & I_{\scriptscriptstyle L}^x & 0 & 0 & 0 \\
0 & 0 & 2G_2^{\scriptscriptstyle X} & 0 & 0 & 0 & 0 & 0 & 0 & I_{\scriptscriptstyle U}^x & 0 & 0 \\
-2D_{\scriptscriptstyle Y}^{-1}J & 0 & 0 & 2D_{\scriptscriptstyle Y}^{-1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 2G_1^{\scriptscriptstyle S} & 0 & 0 & 0 & 0 & 0 & I_{\scriptscriptstyle L}^s & 0 \\
0 & 0 & 0 & 0 & 0 & 2G_2^{\scriptscriptstyle S} & 0 & 0 & 0 & 0 & 0 & I_{\scriptscriptstyle U}^s \\
J & 0 & 0 & -I_m & 0 & 0 & D_{\scriptscriptstyle Y} & 0 & 0 & 0 & 0 & 0 \\
A & 0 & 0 & 0 & 0 & 0 & 0 & D_{\scriptscriptstyle A} & 0 & 0 & 0 & 0 \\
0 & I_{\scriptscriptstyle L}^x & 0 & 0 & 0 & 0 & 0 & 0 & G_1^{\scriptscriptstyle Z} & 0 & 0 & 0 \\
0 & 0 & I_{\scriptscriptstyle U}^x & 0 & 0 & 0 & 0 & 0 & 0 & G_2^{\scriptscriptstyle Z} & 0 & 0 \\
0 & 0 & 0 & 0 & I_{\scriptscriptstyle L}^s & 0 & 0 & 0 & 0 & 0 & G_1^{\scriptscriptstyle W} & 0 \\
0 & 0 & 0 & 0 & 0 & I_{\scriptscriptstyle U}^s & 0 & 0 & 0 & 0 & 0 & G_2^{\scriptscriptstyle W}
\end{pmatrix},$$

where $H_1 = H(x, 2\pi^{\scriptscriptstyle Y} - y) + \frac{2}{\mu^{\scriptscriptstyle A}} A^{\mathrm{T}} A + \frac{2}{\mu^{\scriptscriptstyle P}} J(x)^{\mathrm{T}} J(x)$, and $I_{\scriptscriptstyle L}^x$, $I_{\scriptscriptstyle L}^x$, $I_{\scriptscriptstyle L}^s$, $I_{\scriptscriptstyle U}^s$ are identity matrices of size $n_{\scriptscriptstyle L}$, $n_{\scriptscriptstyle U}$, $m_{\scriptscriptstyle L}$, $m_{\scriptscriptstyle U}$ respectively. In addition

$$\begin{aligned}
G_1^{\scriptscriptstyle X} &= (X_1^{\mu})^{-1}\big(\varPi_1^{\scriptscriptstyle Z} + \mu^{\scriptscriptstyle B} I\big), & G_2^{\scriptscriptstyle X} &= (X_2^{\mu})^{-1}\big(\varPi_2^{\scriptscriptstyle Z} + \mu^{\scriptscriptstyle B} I\big), \\
G_1^{\scriptscriptstyle S} &= (S_1^{\mu})^{-1}\big(\varPi_1^{\scriptscriptstyle W} + \mu^{\scriptscriptstyle B} I\big), & G_2^{\scriptscriptstyle S} &= (S_2^{\mu})^{-1}\big(\varPi_1^{\scriptscriptstyle W} + \mu^{\scriptscriptstyle B} I\big), \\
G_1^{\scriptscriptstyle Z} &= (Z_1^{\mu})^{-1}\big(\varPi_1^{\scriptscriptstyle X} + \mu^{\scriptscriptstyle B} I\big), & G_2^{\scriptscriptstyle Z} &= (Z_2^{\mu})^{-1}\big(\varPi_2^{\scriptscriptstyle X} + \mu^{\scriptscriptstyle B} I\big), \\
G_1^{\scriptscriptstyle W} &= (W_1^{\mu})^{-1}\big(\varPi_1^{\scriptscriptstyle S} + \mu^{\scriptscriptstyle B} I\big), & G_2^{\scriptscriptstyle W} &= (W_2^{\mu})^{-1}\big(\varPi_2^{\scriptscriptstyle S} + \mu^{\scriptscriptstyle B} I\big),
\end{aligned}$$

with $\varPi_1^{\scriptscriptstyle Z} = \operatorname{diag}(\pi_1^{\scriptscriptstyle Z})$, $\varPi_2^{\scriptscriptstyle Z} = \operatorname{diag}(\pi_2^{\scriptscriptstyle Z})$, $\varPi_1^{\scriptscriptstyle W} = \operatorname{diag}(\pi_1^{\scriptscriptstyle W})$, $\varPi_2^{\scriptscriptstyle W} = \operatorname{diag}(\pi_2^{\scriptscriptstyle W})$, $X_1^{\scriptscriptstyle E} = \operatorname{diag}(x_1^{\scriptscriptstyle E})$, $X_2^{\scriptscriptstyle E} = \operatorname{diag}(x_2^{\scriptscriptstyle E})$, $S_1^{\scriptscriptstyle E} = \operatorname{diag}(s_1^{\scriptscriptstyle E})$, $W_1^{\scriptscriptstyle E} = \operatorname{diag}(w_1^{\scriptscriptstyle E})$, $W_2^{\scriptscriptstyle E} = \operatorname{diag}(w_2^{\scriptscriptstyle E})$, $Z_1^{\scriptscriptstyle E} = \operatorname{diag}(z_1^{\scriptscriptstyle E})$ and $Z_2^{\scriptscriptstyle E} = \operatorname{diag}(z_2^{\scriptscriptstyle E})$.

A.5 Derivation of the Primal-Dual Line-Search Direction

The primal-dual penalty-barrier problem may be written in the form

$$\underset{p\in\mathcal{I}}{\text{minimize}}\ \ M(p) \quad \text{subject to} \ \ Cp = b_{C},$$

where

$$\mathcal{I} = \{p : p = (x, x_1, x_2, s, s_1, s_2, y, v, z_1, z_2, w_1, w_2), \ \ \text{with } x_i + \mu^{B} e > 0,\ s_i + \mu^{B} e > 0,\ z_i + \mu^{B} e > 0,\ w_i + \mu^{B} e > 0 \text{ for } i = 1, 2\},$$

and

$$C = \begin{pmatrix} E_X & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E_{L} & -I^x_{L} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E_{U} & 0 & I^x_{U} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_X & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_{L} & -I^s_{L} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_{U} & 0 & I^s_{U} & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad \text{and} \quad b_{C} = \begin{pmatrix} b_X \\ \ell^X \\ u^X \\ h_X \\ \ell^S \\ u^S \end{pmatrix}. \tag{A.10}$$

Let p be any vector in $\mathcal{I}$ such that $Cp = b_{C}$. The Newton direction Δp is given by the solution of the subproblem

$$\underset{\Delta p}{\text{minimize}}\ \ \nabla M(p)^{\mathrm{T}}\Delta p + \tfrac{1}{2}\Delta p^{\mathrm{T}}\nabla^2 M(p)\Delta p \quad \text{subject to} \ \ C\Delta p = b_{C} - Cp = 0. \tag{A.11}$$

Let N denote a matrix whose columns form a basis for null(C), i.e., the columns of N are linearly independent and $CN = 0$. Every feasible direction Δp may be written in the form $\Delta p = Nd$. This implies that d satisfies the reduced equations $N^{\mathrm{T}}\nabla^2 M(p)Nd = -N^{\mathrm{T}}\nabla M(p)$. However, instead of solving (A.11), we formulate a linearly constrained *approximate* Newton method by approximating the Hessian $\nabla^2 M(p)$ by a matrix $B(p)$ such that $N^{\mathrm{T}}B(p)N$ is positive definite with $N^{\mathrm{T}}B(p)N \approx N^{\mathrm{T}}\nabla^2 M(p)N$. Consider the matrix B obtained by replacing π^{Y} by y, π^{Z}_1 by z_1, π^{Z}_2 by z_2, π^{W}_1 by w_1, π^{W}_2 by w_2, x^{E}_1 by x_1, x^{E}_2 by x_2, s^{E}_1 by s_1, s^{E}_2 by s_2, z^{E}_1 by z_1, z^{E}_2 by z_2, w^{E}_1 by w_1 and w^{E}_2 by w_2 in $\nabla^2 M(x,\, x_1,\, x_2,\, s,\, s_1,\, s_2,\, y,\, v,\, z_1,\, z_2,\, w_1,\, w_2)$. This gives an approximate Hessian $B(x,\, x_1,\, x_2,\, s,\, s_1,\, s_2,\, y,\, v,\, z_1,\, z_2,\, w_1,\, w_2)$ of the form

$$\begin{pmatrix}
H^B + \frac{2}{\mu^A}A^{\mathrm{T}}A + \frac{2}{\mu^P}J^{\mathrm{T}}J & 0 & 0 & -2J^{\mathrm{T}}D_Y^{-1} & 0 & 0 & J^{\mathrm{T}} & A^{\mathrm{T}} & 0 & 0 & 0 & 0 \\
0 & 2(D_1^z)^{-1} & 0 & 0 & 0 & 0 & 0 & 0 & I_L^x & 0 & 0 & 0 \\
0 & 0 & -2(D_2^z)^{-1} & 0 & 0 & 0 & 0 & 0 & 0 & I_U^x & 0 & 0 \\
-2D_Y^{-1}J & 0 & 0 & 2D_Y^{-1} & 0 & 0 & -I_m & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 2(D_1^W)^{-1} & 0 & 0 & 0 & 0 & 0 & I_L^s & 0 \\
0 & 0 & 0 & 0 & 0 & 2(D_2^W)^{-1} & 0 & 0 & 0 & 0 & 0 & I_U^s \\
J & 0 & 0 & -I_m & 0 & 0 & D_Y & 0 & 0 & 0 & 0 & 0 \\
A & 0 & 0 & 0 & 0 & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\
0 & I_L^x & 0 & 0 & 0 & 0 & 0 & 0 & D_1^z & 0 & 0 & 0 \\
0 & 0 & I_U^x & 0 & 0 & 0 & 0 & 0 & 0 & D_2^z & 0 & 0 \\
0 & 0 & 0 & 0 & I_L^s & 0 & 0 & 0 & 0 & 0 & D_1^W & 0 \\
0 & 0 & 0 & 0 & 0 & I_U^s & 0 & 0 & 0 & 0 & 0 & D_2^W
\end{pmatrix},$$

where $H^B \approx H(x,y)$ is chosen so that the approximate reduced Hessian $N^{\mathrm{T}}B(p)N$ is positive definite (see Section A.7). Given $B(p)$, an approximate Newton direction is given by the solution of the QP subproblem

$$\underset{\Delta p}{\text{minimize}} \;\; \nabla M(p)^{\mathrm{T}}\Delta p + \tfrac{1}{2}\Delta p^{\mathrm{T}}B(p)\Delta p \quad \text{subject to} \;\; C\Delta p = 0.$$

Let N denote a matrix whose columns form a basis for null(C), i.e., the columns of N are linearly independent and $CN = 0$. Every feasible Δp may be written in the form $\Delta p = Nd$. This implies that d satisfies the reduced equations $N^{\mathrm{T}}B(p)Nd = -N^{\mathrm{T}}\nabla M(p)$.

Consider the null-space basis defined from the columns of

$$N = \begin{pmatrix} E_F^{\mathrm{T}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_F^{\mathrm{T}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I_m & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_A & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_L^x & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_U^x & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I_L^s & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_U^s \end{pmatrix}, \tag{A.12}$$

where $E_{LF} = E_L E_F^{\mathrm{T}}$, $E_{UF} = E_U E_F^{\mathrm{T}}$, $L_{LF} = L_L L_F^{\mathrm{T}}$ and $L_{UF} = L_U L_F^{\mathrm{T}}$. The definition of N of (A.12) gives the reduced Hessian $N^{\mathrm{T}} B(p) N$ such that

$$\begin{pmatrix} \widehat{H}_F & -2J_F^{\mathrm{T}} D_Y^{-1} L_F^{\mathrm{T}} & J_F^{\mathrm{T}} & A_F^{\mathrm{T}} & E_{LF}^{\mathrm{T}} & -E_{UF}^{\mathrm{T}} & 0 & 0 \\ -2L_F D_Y^{-1} J_F & 2L_F (D_Y^{-1} + D_W^{\dagger}) L_F^{\mathrm{T}} & -L_F & 0 & 0 & 0 & L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_F & -L_F^{\mathrm{T}} & D_Y & 0 & 0 & 0 & 0 & 0 \\ A_F & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & D_1^z & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & D_2^z & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & D_1^W & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & D_2^W \end{pmatrix},$$

where $J_F = J(x)E_F^{\mathrm{T}}$, $A_F = AE_F^{\mathrm{T}}$, $\widehat{H}_F = E_F H^B E_F^{\mathrm{T}} + \frac{2}{\mu^A} A_F^{\mathrm{T}} A_F + \frac{2}{\mu^P} J_F^{\mathrm{T}} J_F + 2\big(E_{LF}^{\mathrm{T}}(D_1^Z)^{-1}E_{LF} + E_{UF}^{\mathrm{T}}(D_2^Z)^{-1}E_{UF}\big)$ and $D_W = \big((L_L^{\mathrm{T}}(D_1^W)^{-1}L_L + L_U^{\mathrm{T}}(D_2^W)^{-1}L_U)\big)^{\dagger}$. Similarly, the reduced gradient $N^{\mathrm{T}}\nabla M(p)$ is given by

$$\begin{pmatrix} g_F - A_F^{\mathrm{T}}\big(2\pi^V - v\big) - J_F^{\mathrm{T}}\big(2\pi^Y - y\big) - E_{LF}(2\pi_1^Z - z_1) + E_{UF}(2\pi_2^Z - z_2) \\ 2\pi_F^Y - y_F - L_{LF}(2\pi_1^W - w_1) + L_{UF}(2\pi_2^W - w_2) \\ -D_Y(\pi^Y - y) \\ -D_A(\pi^V - v) \\ -D_1^Z(\pi_1^Z - z_1) \\ -D_2^Z(\pi_2^Z - z_2) \\ -D_1^W(\pi_1^W - w_1) \\ -D_2^W(\pi_2^W - w_2) \end{pmatrix},$$

where $g_F = E_F \nabla f(x)$, $\pi_F^Y = L_F \pi^Y$ and $y_F = L_F y$. The reduced approximate Newton equations $N^{\mathrm{T}} B(p) N d = -N^{\mathrm{T}} \nabla M(p)$ are then

$$\begin{pmatrix} \widehat{H}_F & -2J_F^{\mathrm{T}} D_Y^{-1} L_F^{\mathrm{T}} & J_F^{\mathrm{T}} & A_F^{\mathrm{T}} & E_{LF}^{\mathrm{T}} & -E_{UF}^{\mathrm{T}} & 0 & 0 \\ -2L_F D_Y^{-1} J_F & 2L_F(D_Y^{-1} + D_W^{\dagger}) L_F^{\mathrm{T}} & -L_F & 0 & 0 & 0 & L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_F & -L_F^{\mathrm{T}} & D_Y & 0 & 0 & 0 & 0 & 0 \\ A_F & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & D_1^Z & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & D_2^Z & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & D_1^W & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & D_2^W \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \\ d_7 \\ d_8 \end{pmatrix}$$

$$= - \begin{pmatrix} g_F - A_F^{\mathrm{T}}\big(2\pi^V - v\big) - J_F^{\mathrm{T}}\big(2\pi^Y - y\big) - E_{LF}(2\pi_1^Z - z_1) + E_{UF}(2\pi_2^Z - z_2) \\ 2\pi_F^Y - y_F - L_{LF}(2\pi_1^W - w_1) + L_{UF}(2\pi_2^W - w_2) \\ -D_Y(\pi^Y - y) \\ -D_A(\pi^V - v) \\ -D_1^Z(\pi_1^Z - z_1) \\ -D_2^Z(\pi_2^Z - z_2) \\ -D_1^W(\pi_1^W - w_1) \\ -D_2^W(\pi_2^W - w_2) \end{pmatrix}. \qquad \text{(A.13)}$$

Given any nonsingular matrix R, the direction d satisfies $RN^{\mathrm{T}}B(p)Nd = -RN^{\mathrm{T}}\nabla M(p)$. In particular, consider the block upper-triangular matrix R such that

$$R = \begin{pmatrix} I_F^x & 0 & -2J_F^{\mathrm{T}}D_Y^{-1} & -2A_F^{\mathrm{T}}D_A^{-1} & -2E_{LF}^{\mathrm{T}}(D_1^z)^{-1} & 2E_{UF}^{\mathrm{T}}(D_2^z)^{-1} & 0 & 0 \\ & I_F^s & 2L_F D_Y^{-1} & 0 & 0 & 0 & -2L_{LF}^{\mathrm{T}}(D_1^W)^{-1} & 2L_{UF}^{\mathrm{T}}(D_2^W)^{-1} \\ & & I_m & 0 & 0 & 0 & 0 & 0 \\ & & & I_A & 0 & 0 & 0 & 0 \\ & & & & I_L^x & 0 & 0 & 0 \\ & & & & & I_U^x & 0 & 0 \\ & & & & & & I_L^s & 0 \\ & & & & & & & I_U^s \end{pmatrix},$$

where again, I_L^x, I_U^x, I_L^s, I_U^s are identity matrices of size n_L, n_U, m_L, and m_U respectively. Then R is nonsingular with

$$RN^{\mathrm{T}}B(p)N = \begin{pmatrix} E_F H^B E_F^{\mathrm{T}} & 0 & -J_F^{\mathrm{T}} & -A_F^{\mathrm{T}} & -E_{LF}^{\mathrm{T}} & E_{UF}^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & L_F & 0 & 0 & 0 & -L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_F & -L_F^{\mathrm{T}} & D_Y & 0 & 0 & 0 & 0 & 0 \\ A_F & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & D_1^z & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & D_2^z & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & D_1^W & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & D_2^W \end{pmatrix}$$

Also,

$$RN^{\mathrm{T}}\nabla M(p) = \begin{pmatrix} g_{F} - J_{F}^{\mathrm{T}} y - A_{F}^{\mathrm{T}} v - z_1 + z_2 \\ y_{F} - w_1 + w_2 \\ -D_{Y}(\pi^{Y} - y) \\ -D_{A}(\pi^{V} - v) \\ -D_1^{Z}(\pi_1^{Z} - z_1) \\ -D_2^{Z}(\pi_2^{Z} - z_2) \\ -D_1^{W}(\pi_1^{W} - w_1) \\ -D_2^{W}(\pi_2^{W} - w_2) \end{pmatrix}.$$

This gives the following (unsymmetric) reduced approximate Newton equations for d:

$$\begin{pmatrix} E_{F} H^{B} E_{F}^{\mathrm{T}} & 0 & -J_{F}^{\mathrm{T}} & -A_{F}^{\mathrm{T}} & -E_{LF}^{\mathrm{T}} & E_{UF}^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & L_{F} & 0 & 0 & 0 & -L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_{F} & -L_{F}^{\mathrm{T}} & D_{Y} & 0 & 0 & 0 & 0 & 0 \\ A_{F} & 0 & 0 & D_{A} & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & D_1^{Z} & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & D_2^{Z} & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & D_1^{W} & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & D_2^{W} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \\ d_7 \\ d_8 \end{pmatrix} = - \begin{pmatrix} g_{F} - J_{F}^{\mathrm{T}} y - A_{F}^{\mathrm{T}} v - E_{LF}^{\mathrm{T}} z_1 + E_{UF}^{\mathrm{T}} z_2 \\ y_{F} - L_{LF}^{\mathrm{T}} w_1 + L_{UF}^{\mathrm{T}} w_2 \\ -D_{Y}(\pi^{Y} - y) \\ -D_{A}(\pi^{V} - v) \\ -D_1^{Z}(\pi_1^{Z} - z_1) \\ -D_2^{Z}(\pi_2^{Z} - z_2) \\ -D_1^{W}(\pi_1^{W} - w_1) \\ -D_2^{W}(\pi_2^{W} - w_2) \end{pmatrix}. \tag{A.14}$$

Then, the identity $\varDelta p = Nd$ implies that

$$\varDelta p = \begin{pmatrix} \varDelta x \\ \varDelta x_1 \\ \varDelta x_2 \\ \varDelta s \\ \varDelta s_1 \\ \varDelta s_2 \\ \varDelta y \\ \varDelta v \\ \varDelta z_1 \\ \varDelta z_2 \\ \varDelta w_1 \\ \varDelta w_2 \end{pmatrix} = Nd = \begin{pmatrix} E_F^{\mathrm{T}} d_1 \\ d_1 \\ -d_1 \\ L_F^{\mathrm{T}} d_2 \\ d_2 \\ -d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \\ d_7 \\ d_8 \end{pmatrix}. \tag{A.15}$$

These identities allow us to write equations (A.14) in the form

$$\begin{pmatrix} E_F H^B E_F^{\mathrm{T}} & 0 & -J_F^{\mathrm{T}} & -A_F^{\mathrm{T}} & -E_{LF}^{\mathrm{T}} & E_{UF}^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & L_F & 0 & 0 & 0 & -L_{LF}^{\mathrm{T}} & L_{UF}^{\mathrm{T}} \\ J_F & -L_F^{\mathrm{T}} & D_Y & 0 & 0 & 0 & 0 & 0 \\ A_F & 0 & 0 & D_A & 0 & 0 & 0 & 0 \\ E_{LF} & 0 & 0 & 0 & D_1^Z & 0 & 0 & 0 \\ -E_{UF} & 0 & 0 & 0 & 0 & D_2^Z & 0 & 0 \\ 0 & L_{LF} & 0 & 0 & 0 & 0 & D_1^W & 0 \\ 0 & -L_{UF} & 0 & 0 & 0 & 0 & 0 & D_2^W \end{pmatrix} \begin{pmatrix} \Delta x_F \\ \Delta s_F \\ \Delta y \\ \Delta v \\ \Delta z_1 \\ \Delta z_2 \\ \Delta w_1 \\ \Delta w_2 \end{pmatrix} = - \begin{pmatrix} g_F - J_F^{\mathrm{T}} y - A_F^{\mathrm{T}} v - E_{LF}^{\mathrm{T}} z_1 + E_{UF}^{\mathrm{T}} z_2 \\ y_F - L_{LF}^{\mathrm{T}} w_1 + L_{UF}^{\mathrm{T}} w_2 \\ -D_Y(\pi^Y - y) \\ -D_A(\pi^V - v) \\ -D_1^Z(\pi_1^Z - z_1) \\ -D_2^Z(\pi_2^Z - z_2) \\ -D_1^W(\pi_1^W - w_1) \\ -D_2^W(\pi_2^W - w_2) \end{pmatrix}, \tag{A.16}$$

with $\Delta x = E_F^{\mathrm{T}} \Delta x_F$, $\Delta s = L_F^{\mathrm{T}} \Delta s_F$, $\Delta x_1 = \Delta x_F - (\ell^X - E_L x + x_1)$, $\Delta x_2 = -\Delta x_F + (u^X - E_U x - x_2)$, $\Delta s_1 = \Delta s_F - (\ell^S - L_L s + s_1)$ and $\Delta s_2 = -\Delta s_F + (u^S - L_U s - s_2)$.

The shifted penalty-barrier equations (A.16) are the same as the path-following equations (A.8) except for the $(1,1)$ block, where H_F is replaced by $E_F H^B E_F^{\mathrm{T}}$.

A.6 The Shifted Primal-Dual Penalty-Barrier Direction

In this section we consider the solution of the shifted primal-dual penalty-barrier equations (A.16). Collecting terms and reordering the equations and unknowns, we obtain

$$\begin{pmatrix} D_A & 0 & 0 & 0 & 0 & 0 & A_F & 0 \\ 0 & D_1^z & 0 & 0 & 0 & 0 & E_{LF} & 0 \\ 0 & 0 & D_2^z & 0 & 0 & 0 & -E_{UF} & 0 \\ 0 & 0 & 0 & D_1^W & 0 & L_{LF} & 0 & 0 \\ 0 & 0 & 0 & 0 & D_2^W & -L_{UF} & 0 & 0 \\ 0 & 0 & 0 & -L_{LF}^T & L_{UF}^T & 0 & 0 & L_F \\ -A_F^T & -E_{LF}^T & E_{UF}^T & 0 & 0 & 0 & E_F H^B E_F^T & -J_F^T \\ 0 & 0 & 0 & 0 & 0 & -L_F^T & J_F & D_Y \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta z_1 \\ \Delta z_2 \\ \Delta w_1 \\ \Delta w_2 \\ \Delta s_F \\ \Delta x_F \\ \Delta y \end{pmatrix} = - \begin{pmatrix} D_A(v - \pi^V) \\ D_1^z(z_1 - \pi_1^z) \\ D_2^z(z_2 - \pi_2^z) \\ D_1^W(w_1 - \pi_1^W) \\ D_2^W(w_2 - \pi_2^W) \\ y_F - L_{LF}^T w_1 + L_{UF}^T w_2 \\ g_F - J_F^T y - A_F^T v - E_{LF}^T z_1 + E_{UF}^T z_2 \\ D_Y(y - \pi^Y) \end{pmatrix}. \tag{A.17}$$

Consider the diagonal matrices

$$D_W = \left(L_L^T (D_1^W)^{-1} L_L + L_U^T (D_2^W)^{-1} L_U\right)^\dagger \quad \text{and} \quad D_Z = \left(E_L^T (D_1^z)^{-1} E_L + E_U^T (D_2^z)^{-1} E_U\right)^\dagger,$$

where $(\,\cdot\,)^\dagger$ denotes the Moore-Penrose pseudoinverse of a matrix. The identity $I_m = L_X^T L_X + L_F^T L_F$ implies that the $m \times m$ matrix D_W satisfies the identities

$$L_F^T L_F D_W = D_W = D_W L_F^T L_F, \quad \text{and} \quad L_X^T L_X D_W = 0.$$

If equations (A.17) are premultiplied by the matrix

$$\begin{pmatrix} I_A &&&&&&& \\ 0 & I_L^x &&&&&& \\ 0 & 0 & I_U^x &&&&& \\ 0 & 0 & 0 & I_L^s &&&& \\ 0 & 0 & 0 & 0 & I_U^s &&& \\ 0 & 0 & 0 & L_{LF}^{\mathrm{T}}(D_1^W)^{-1} & -L_{UF}^{\mathrm{T}}(D_2^W)^{-1} & I_F^s && \\ A_F^{\mathrm{T}} D_A^{-1} & E_{LF}^{\mathrm{T}}(D_1^Z)^{-1} & -E_{UF}^{\mathrm{T}}(D_2^Z)^{-1} & 0 & 0 & 0 & I_F^x & \\ 0 & 0 & 0 & D_W L_L^{\mathrm{T}}(D_1^W)^{-1} & -D_W L_U^{\mathrm{T}}(D_2^W)^{-1} & L_F^{\mathrm{T}} D_W & 0 & I_m \end{pmatrix}$$

gives the block upper-triangular system

$$\begin{pmatrix} D_A & 0 & 0 & 0 & 0 & 0 & A_F & 0 \\ 0 & D_1^Z & 0 & 0 & 0 & 0 & E_{LF} & 0 \\ 0 & 0 & D_2^Z & 0 & 0 & 0 & -E_{UF} & 0 \\ 0 & 0 & 0 & D_1^W & 0 & L_{LF} & 0 & 0 \\ 0 & 0 & 0 & 0 & D_2^W & -L_{UF} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & L_F D_W^{\dagger} L_F^{\mathrm{T}} & 0 & L_F \\ 0 & 0 & 0 & 0 & 0 & 0 & \widetilde{H}_F & -J_F^{\mathrm{T}} \\ 0 & 0 & 0 & 0 & 0 & 0 & J_F & D_Y + D_W \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta z_1 \\ \Delta z_2 \\ \Delta w_1 \\ \Delta w_2 \\ \Delta s_F \\ \Delta x_F \\ \Delta y \end{pmatrix} = - \begin{pmatrix} D_A(v - \pi^V) \\ D_1^Z(z_1 - \pi_1^Z) \\ D_2^Z(z_2 - \pi_2^Z) \\ D_1^W(w_1 - \pi_1^W) \\ D_2^W(w_2 - \pi_2^W) \\ y_F - \pi_F^W \\ g_F - J_F^{\mathrm{T}} y - A_F^{\mathrm{T}} \pi^V - \pi_F^Z \\ D_W\big(y_F - \pi_F^W\big) + D_Y\big(y - \pi^Y\big) \end{pmatrix},$$

where $\widetilde{H}_F = E_F H^B E_F^{\mathrm{T}} + A_F^{\mathrm{T}} D_A^{-1} A_F + E_F D_Z^{\dagger} E_F^{\mathrm{T}}$, $\pi_F^W = L_{LF}^{\mathrm{T}} \pi_1^W - L_{UF}^{\mathrm{T}} \pi_2^W$ and $\pi_F^Z = E_{LF}^{\mathrm{T}} \pi_1^Z - E_{UF}^{\mathrm{T}} \pi_2^Z$. Using block back-substitution, Δx_F and Δy can be computed by solving the equations

$$\begin{pmatrix} \widetilde{H}_F & -J_F^{\mathrm{T}} \\ J_F & D_Y + D_W \end{pmatrix} \begin{pmatrix} \Delta x_F \\ \Delta y \end{pmatrix} = - \begin{pmatrix} g_F - J_F^{\mathrm{T}} y - A_F^{\mathrm{T}} \pi^V - \pi_F^Z \\ D_W\big(y - \pi^W\big) + D_Y\big(y - \pi^Y\big) \end{pmatrix}.$$

Once Δx_F and Δy are known, the full vector Δx is computed as $\Delta x = E_F^{\mathrm{T}} \Delta x_F$. Using the identity $\Delta s = L_F^{\mathrm{T}} \Delta s_F$ in the sixth block of equations gives

$$\Delta s = -D_W(y + \Delta y - \pi^W).$$

There are several ways of computing Δw_1 and Δw_2. Instead of using the block upper-triangular system above, we use the last two blocks of equations of (A.8) to give

$$\Delta w_1 = -(S_1^\mu)^{-1}\big(w_1 \cdot (L_L(s + \Delta s) - \ell^S + \mu^B e) - \mu^B w_1^E + \mu^B L_L(s - s^E + \Delta s)\big),$$

and

$$\Delta w_2 = -(S_2^\mu)^{-1}\big(w_2 \cdot (u^S - L_U(s + \Delta s) + \mu^B e) - \mu^B w_2^E + \mu^B L_U(s^E - s - \Delta s)\big).$$

Similarly, using (A.8) to solve for Δz_1 and Δz_2 yields

$$\Delta z_1 = -(X_1^\mu)^{-1}\big(z_1 \cdot (E_L(x + \Delta x) - \ell^X + \mu^B e) - \mu^B z_1^E + \mu^B E_L(x - x^E + \Delta x)\big),$$

and

$$\Delta z_2 = -(X_2^\mu)^{-1}\big(z_2 \cdot (u^X - E_U(x + \Delta x) + \mu^B e) - \mu^B z_2^E + \mu^B E_U(x^E - x - \Delta x)\big).$$

Similarly, using the first block of equations (A.17) to solve for Δv gives $\Delta v = -(v - \widehat{\pi}^V)$, with $\widehat{\pi}^V = v^E - \frac{1}{\mu^A}\big(A(x + \Delta x) - b\big)$. Finally, the vectors Δw_X and Δz_X are recovered as $\Delta w_X = [\,y + \Delta y - w\,]_X$ and $\Delta z_X = [\,g + H\Delta x - J^{\mathrm{T}}(y + \Delta y) - z\,]_X$, where $w = L_X^{\mathrm{T}} w_X + L_L^{\mathrm{T}} w_1 - L_U^{\mathrm{T}} w_2$ and $z = E_X^{\mathrm{T}} z_X + E_L^{\mathrm{T}} z_1 - E_U^{\mathrm{T}} z_2$.

A.7 Summary

The results of the preceding section imply that the solution of the path-following equations $F'(v_P)\varDelta v_P = -F(v_P)$ with F and F' given by (A.5) and (A.6) may be computed as follows. Let x and s be given primal variables and slack variables such that $E_X x = b_X$, $L_X s = h_X$ with $\ell^X - \mu^B < E_L x$, $E_U x < u^X + \mu^B$, $\ell^S - \mu^B < L_L s$, $L_U s < u^S + \mu^B$. Similarly, let z_1, z_2, w_1, w_2 and y denote dual variables such that $w_1 > 0$, $w_2 > 0$, $z_1 > 0$, and $z_2 > 0$. Consider the diagonal matrices $X_1^\mu = \mathrm{diag}(E_L x - \ell^X + \mu^B e)$, $X_2^\mu = \mathrm{diag}(u^X - E_U x + \mu^B e)$, $Z_1 = \mathrm{diag}(z_1)$, $Z_2 = \mathrm{diag}(z_2)$, $W_1 = \mathrm{diag}(w_1)$, $W_2 = \mathrm{diag}(w_2)$, $S_1^\mu = \mathrm{diag}(L_L s - \ell^S + \mu^B e)$ and $S_2^\mu = \mathrm{diag}(u^S - L_U s + \mu^B e)$. Consider the quantities

$$\begin{aligned}
D_Y &= \mu^P I_m, & \pi^Y &= y^E - \frac{1}{\mu^P}(c - s),\\
D_A &= \mu^A I_A, & \pi^V &= v^E - \frac{1}{\mu^A}(Ax - b),\\
(D_1^Z)^{-1} &= (X_1^\mu)^{-1} Z_1^\mu, & (D_1^W)^{-1} &= (S_1^\mu)^{-1} W_1^\mu,\\
(D_2^Z)^{-1} &= (X_2^\mu)^{-1} Z_2^\mu, & (D_2^W)^{-1} &= (S_2^\mu)^{-1} W_2^\mu,\\
D_Z &= \big(E_L^{\mathrm{T}} (D_1^Z)^{-1} E_L + E_U^{\mathrm{T}} (D_2^Z)^{-1} E_U\big)^\dagger, & D_W &= \big(L_L^{\mathrm{T}} (D_1^W)^{-1} L_L + L_U^{\mathrm{T}} (D_2^W)^{-1} L_U\big)^\dagger,\\
\pi_1^Z &= \mu^B (X_1^\mu)^{-1}(z_1^E - x_1 + x_1^E), & \pi_1^W &= \mu^B (S_1^\mu)^{-1}(w_1^E - s_1 + s_1^E),\\
\pi_2^Z &= \mu^B (X_2^\mu)^{-1}(z_2^E - x_2 + x_2^E), & \pi_2^W &= \mu^B (S_2^\mu)^{-1}(w_2^E - s_2 + s_2^E),\\
\pi^Z &= E_L^{\mathrm{T}} \pi_1^Z - E_U^{\mathrm{T}} \pi_2^Z, & \pi^W &= L_L^{\mathrm{T}} \pi_1^W - L_U^{\mathrm{T}} \pi_2^W.
\end{aligned}$$

Choose H_F^B so that H_F^B approximates $E_F H(x,y) E_F^{\mathrm{T}}$ and the KKT matrix

$$\begin{pmatrix} H_F^B + A_F^{\mathrm{T}} D_A^{-1} A_F + E_F D_Z^\dagger E_F^{\mathrm{T}} & J_F^{\mathrm{T}} \\ J_F & -(D_Y + D_W) \end{pmatrix}$$

is nonsingular with m negative eigenvalues. (A common choice of H_F^B is the matrix $E_F\big(H(x,y)+\sigma I_n\big)E_F^{\mathrm{T}}$ for some nonnegative scalar σ.) Solve the KKT system

$$\begin{pmatrix} H_F^B + A_F^{\mathrm{T}} D_A^{-1} A_F + E_F D_Z^{\dagger} E_F^{\mathrm{T}} & -J_F^{\mathrm{T}} \\ J_F & D_Y + D_W \end{pmatrix} \begin{pmatrix} \Delta x_F \\ \Delta y \end{pmatrix} = - \begin{pmatrix} g_F - J_F^{\mathrm{T}} y - A_F^{\mathrm{T}} \pi^V - \pi_F^Z \\ D_W\big(y_F - \pi_F^W\big) + D_Y\big(y - \pi^Y\big) \end{pmatrix},$$

and set

$$\begin{array}{lll}
\Delta x = E_F^{\mathrm{T}} \Delta x_F, & \widehat{x} = x + \Delta x, & \Delta z_1 = -(X_1^{\mu})^{-1}\big(z_1 \cdot (E_L\widehat{x} - \ell^X + \mu^B e) - \mu^B z_1^E + \mu^B E_L(x - x^E + \Delta x)\big), \\
& & \Delta z_2 = -(X_2^{\mu})^{-1}\big(z_2 \cdot (u^X - E_U\widehat{x} + \mu^B e) - \mu^B z_2^E + \mu^B E_U(x^E - x - \Delta x)\big), \\
& \widehat{y} = y + \Delta y, & \Delta s \ = -D_W(\widehat{y} - \pi^W), \\
& \widehat{s} = s + \Delta s, & \Delta w_1 = -(S_1^{\mu})^{-1}\big(w_1 \cdot (L_L\widehat{s} - \ell^S + \mu^B e) - \mu^B w_1^E + \mu^B L_L(s - s^E + \Delta s)\big), \\
& & \Delta w_2 = -(S_2^{\mu})^{-1}\big(w_2 \cdot (u^S - L_U\widehat{s} + \mu^B e) - \mu^B w_2^E + \mu^B L_U(s^E - s - \Delta s)\big), \\
& \widehat{\pi}^V = v^E - \dfrac{1}{\mu^A}(A\widehat{x} - b), & \Delta v \ = \ \widehat{\pi}^V - v, \\
& w = L_X^{\mathrm{T}} w_X + L_L^{\mathrm{T}} w_1 - L_U^{\mathrm{T}} w_2, & z = \ E_X^{\mathrm{T}} z_X + E_L^{\mathrm{T}} z_1 - E_U^{\mathrm{T}} z_2, \\
& \widehat{v} = v + \Delta v, & \Delta w_X = \ [\,\widehat{y} - w\,]_X, \\
& & \Delta z_X = \ [\,\nabla f(x) + H(x)\Delta x - J(x)^{\mathrm{T}}\widehat{y} - A^{\mathrm{T}}\widehat{v} - z\,]_X.
\end{array}$$

The associated merit function (A.9) can be written as

$$
\begin{aligned}
f(x) - \big(c(x) - s\big)^{\mathrm{T}} y^{E} + \frac{1}{2\mu^{P}}\|c(x) - s\|^2 + \frac{1}{2\mu^{P}}\|c(x) - s + \mu^{P}(y - y^{E})\|^2 \\
- (Ax - b)^{\mathrm{T}} v^{E} + \frac{1}{2\mu^{A}}\|Ax - b\|^2 + \frac{1}{2\mu^{A}}\|Ax - b + \mu^{A}(v - v^{E})\|^2 \\
- \sum_{j=1}^{n_L} \Big\{ \mu^{B}([\,z_1^{E}\,]_j + [\,E_L x^{E} - \ell^{X}\,]_j + \mu^{B}) \ln\big([\,z_1 + \mu^{B} e\,]_j [\,E_L x - \ell^{X} + \mu^{B} e\,]_j^2\big) - [\,z_1 \cdot (E_L x - \ell^{X} + \mu^{B} e)\,]_j - 2\mu^{B}[\,E_L x - \ell^{X}\,]_j \Big\} \\
- \sum_{j=1}^{n_U} \Big\{ \mu^{B}([\,z_2^{E}\,]_j + [\,u^{X} - E_U x^{E}\,]_j + \mu^{B}) \ln\big([\,z_2 + \mu^{B} e\,]_j [\,u^{X} - E_U x + \mu^{B} e\,]_j^2\big) - [\,z_2 \cdot (u^{X} - E_U x + \mu^{B} e)\,]_j - 2\mu^{B}[\,u^{X} - E_U x\,]_j \Big\} \\
- \sum_{i=1}^{m_L} \Big\{ \mu^{B}([\,w_1^{E}\,]_i + [\,L_L s^{E} - \ell^{S}\,]_i + \mu^{B}) \ln\big([\,w_1 + \mu^{B}\,]_i [\,L_L s - \ell^{S} + \mu^{B} e\,]_i^2\big) - [\,w_1 \cdot (L_L s - \ell^{S} + \mu^{B} e)\,]_i - 2\mu^{B}[\,L_L s - \ell^{S}\,]_i \Big\} \\
- \sum_{i=1}^{m_U} \Big\{ \mu^{B}([\,w_2^{E}\,]_i + [\,u^{S} - L_U s^{E}\,] + \mu^{B}) \ln\big([\,w_2 + \mu^{B}\,]_i [\,u^{S} - L_U s + \mu^{B} e\,]_i^2\big) - [\,w_2 \cdot (u^{S} - L_U s + \mu^{B} e)\,]_i - 2\mu^{B}[\,u^{S} - L_U s\,]_i \Big\}.
\end{aligned}
$$

Appendix A, in part, reprints the material by Philip E. Gill and Minxin Zhang, "Equations for a Projected-Search Path-Following Method for Nonlinear Optimization." Center for Computational Mathematics Report CCoM 22-02, Center for Computational Mathematics, University of California San Diego, La Jolla, CA, 2022. The dissertation author was the primary author of this material.

Bibliography

[1] Roberto Andreani, José Mario Martínez, Alberto Ramos, and Paulo J. S. Silva. Strict constraint qualifications and sequential optimality conditions for constrained optimization. *Math. Oper. Res.*, 43(3):693–717, 2018.

[2] Roberto Andreani, José Mario Martínez, and B. F. Svaiter. A new sequential optimality condition for constrained optimization and algorithmic consequences. *SIAM J. Optim.*, 20(6):3533–3554, 2010.

[3] Larry Armijo. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16:1–3, 1966.

[4] Dimitri P. Bertsekas. On the Goldstein-Levitin-Polyak gradient projection method. *IEEE Trans. Automatic Control*, AC-21(2):174–184, 1976.

[5] Dimitri P. Bertsekas. *Constrained optimization and Lagrange multiplier methods.* Computer Science and Applied Mathematics. Academic Press Inc. [Harcourt Brace Jovanovich Publishers], New York, 1982.

[6] Dimitri P. Bertsekas. Projected Newton methods for optimization problems with simple constraints. *SIAM J. Control Optim.*, 20(2):221–246, 1982.

[7] I. Bongartz, A. R. Conn, N. I. M. Gould, and Philippe L. Toint. CUTE: Constrained and unconstrained testing environment. *ACM Trans. Math. Software*, 21(1):123–160, 1995.

[8] Marc G. Breitfeld and David F. Shanno. Computational experience with penalty-barrier methods for nonlinear programming. *Ann. Oper. Res.*, 62:439–463, 1996.

[9] Richard P. Brent. *Algorithms for minimization without derivatives.* Prentice-Hall Inc., Englewood Cliffs, N.J., 1973. Prentice-Hall Series in Automatic Computation.

[10] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.*, 16:1190–1208, 1995.

[11] Richard H. Byrd, Jean Charles Gilbert, and Jorge Nocedal. A trust region method based on interior point techniques for nonlinear programming. *Math. Program.*, 89(1, Ser. A):149–185, 2000.

[12] Richard H. Byrd, Mary E. Hribar, and Jorge Nocedal. An interior point algorithm for large-scale nonlinear programming. *SIAM J. Optim.*, 9(4):877–900, 1999.

[13] Paul H. Calamai and Jorge J. Moré. Projected gradient methods for linearly constrained problems. *Math. Program.*, 39:93–116, 1987.

[14] Andrew R. Conn, Nicholas I. M. Gould, Dominique Orban, and Philippe L. Toint. A primal-dual trust-region algorithm for non-convex nonlinear programming. *Math. Program.*, 87(2, Ser. B):215–249, 2000. Studies in algorithmic optimization.

[15] Andrew R. Conn, Nicholas I. M. Gould, and Philippe L. Toint. A globally convergent Lagrangian barrier algorithm for optimization with general inequality constraints and simple bounds. *Math. Comput.*, 66(217):261–288, 1997.

[16] Andrew R. Conn, Nicholas I. M. Gould, and Philippe L. Toint. A primal-dual algorithm for minimizing a non-convex function subject to bound and linear equality constraints. In *Nonlinear optimization and related topics (Erice, 1998)*, volume 36 of *Appl. Optim.*, pages 15–49. Kluwer Acad. Publ., Dordrecht, 2000.

[17] Andrew R. Conn, Nicholas I. M. Gould, and Philippe L. Toint. *Trust-Region Methods.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[18] Frank E. Curtis and Jorge Nocedal. Flexible penalty functions for nonlinear constrained optimization. *IMA J. Numer. Anal.*, 28(4):749–769, 2008.

[19] J. W. Daniel, W. B. Gragg, L. Kaufman, and G. W. Stewart. Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization. *Math. Comput.*, 30:772–795, 1976.

[20] John E. Dennis, Jr. and Robert B. Schnabel. A new derivation of symmetric positive definite secant updates. In *Nonlinear Programming, 4 (Proc. Sympos., Special Interest Group on Math. Programming, Univ. Wisconsin, Madison, Wis., 1980)*, pages 167–199. Academic Press, New York, 1981.

[21] John E. Dennis, Jr. and Robert B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996. Corrected reprint of the 1983 original.

[22] J. E. Dennis, Jr., M. Heinkenschloss, and L. N. Vicente. Trust-region interior-point SQP algorithms for a class of nonlinear programming problems. *SIAM J. Control Optim.*, 36:1750–1794, 1998.

[23] J. E. Dennis Jr. and J. J. Moré. A characterization of superlinear convergence and its application to quasi-Newton methods. *Math. Comput.*, 28:549–560, 1974.

[24] Elizabeth D. Dolan and Jorge J. Moré. Benchmarking optimization software with performance profiles. *Math. Program.*, 91(2, Ser. A):201–213, 2002.

[25] Iain S. Duff. MA57—a code for the solution of sparse symmetric definite and indefinite systems. *ACM Trans. Math. Software*, 30(2):118–144, 2004.

[26] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang. On the formulation and theory of the Newton interior-point method for nonlinear programming. *J. Optim. Theory Appl.*, 89(3):507–541, 1996.

[27] M. C. Fenelon. *Preconditioned Conjugate-Gradient-Type Methods for Large-Scale Unconstrained Optimization.* PhD thesis, Department of Operations Research, Stanford University, Stanford, CA, 1981.

[28] Michael W. Ferry. *Projected-Search Methods for Box-Constrained Optimization.* PhD thesis, Department of Mathematics, University of California, San Diego, May 2011.

[29] Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang. Supplementary numerical results for projected-search methods for bound-constrained optimization. Center for Computational Mathematics Report CCoM 20-08, Center for Computational Mathematics, University of California, San Diego, La Jolla, CA, 2020.

[30] Michael W. Ferry, Philip E. Gill, Elizabeth Wong, and Minxin Zhang. A limited-memory reduced-Hessian method for bound-constrained optimization. Center for Computational Mathematics Report CCoM 21-01, Center for Computational Mathematics, University of California, San Diego, La Jolla, CA, 2021.

[31] Anthony V. Fiacco and Garth P. McCormick. *Nonlinear Programming.* Classics in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 1990. Reprint of the 1968 original.

[32] Roger Fletcher and Michael J. D. Powell. A rapidly convergent descent method for minimization. *Computer Journal*, 6:163–168, 1963.

[33] Anders Forsgren. Inertia-controlling factorizations for optimization algorithms. *Appl. Numer. Math.*, 43:91–107, 2002.

[34] Anders Forsgren and Philip E. Gill. Primal-dual interior methods for nonconvex nonlinear programming. *SIAM J. Optim.*, 8:1132–1152, 1998.

[35] Anders Forsgren, Philip E. Gill, and Walter Murray. Computing modified Newton directions using a partial Cholesky factorization. *SIAM J. Sci. Comput.*, 16:139–150, 1995.

[36] Anders Forsgren, Philip E. Gill, and Joseph R. Shinnerl. Stability of symmetric ill-conditioned systems arising in interior methods for constrained optimization. *SIAM J. Matrix Anal. Appl.*, 17:187–211, 1996.

[37] Anders Forsgren, Philip E. Gill, and Margaret H. Wright. Interior methods for nonlinear optimization. *SIAM Rev.*, 44:525–597, 2002.

[38] Anders Forsgren and Walter Murray. Newton methods for large-scale linear equality-constrained minimization. *SIAM J. Matrix Anal. Appl.*, 14:560–587, 1993.

[39] Robert Fourer and Sanjay Mehrotra. Performance of an augmented system approach for solving least-squares problems in an interior-point method for linear programming. *Math. Program.*, 19:26–31, 1991.

[40] Robert Fourer and Sanjay Mehrotra. Solving symmetric indefinite systems in an interior-point method for linear programming. *Math. Programming*, 62(1, Ser. B):15–39, 1993.

[41] Robert M. Freund. Theoretical efficiency of a shifted-barrier-function algorithm for linear programming. *Linear Algebra Appl.*, 152:19–41, 1991.

[42] David M. Gay, Michael L. Overton, and Margaret H. Wright. A primal-dual interior method for nonconvex nonlinear programming. In Ya-Xiang Yuan, editor, *Advances in Nonlinear Programming (Beijing, 1996)*, pages 31–56. Kluwer Acad. Publ., Dordrecht, 1998.

[43] E. Michael Gertz and Philip E. Gill. A primal-dual trust-region algorithm for nonlinear programming. Numerical Analysis Report NA 02-1, University of California, San Diego, 2002.

[44] E. Michael Gertz and Philip E. Gill. A primal-dual trust-region algorithm for nonlinear programming. *Math. Program., Ser. B*, 100:49–94, 2004.

[45] N. Ghosh and William W. Hager. A derivative-free bracketing scheme for univariate minimization. *Computers Math. Applic.*, 20(2):23–24, 1990.

[46] Philip E. Gill, Vyacheslav Kungurtsev, and Daniel P. Robinson. A shifted primal-dual penalty-barrier method for nonlinear optimization. *SIAM J. Optim.*, 30(2):1067–1093, 2020.

[47] Philip E. Gill and Michael W. Leonard. Limited-memory reduced-Hessian methods for large-scale unconstrained optimization. *SIAM J. Optim.*, 14:380–401, 2003.

[48] Philip E. Gill, Walter Murray, Dulce B. Ponceleón, and Michael A. Saunders. Primal-dual methods for linear programming. Report SOL 91-3, Department of Operations Research, Stanford University, Stanford, CA, 1991.

[49] Philip E. Gill, Walter Murray, Dulce B. Ponceleón, and Michael A. Saunders. Preconditioners for indefinite systems arising in optimization. *SIAM J. Matrix Anal. Appl.*, 13:292–311, 1992.

[50] Philip E. Gill, Walter Murray, and Michael A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM J. Optim.*, 12(4):979–1006, 2002.

[51] Philip E. Gill, Walter Murray, Michael A. Saunders, and Margaret H. Wright. A note on a sufficient-decrease criterion for a nonderivative step-length procedure. *Math. Programming*, 23(3):349–352, 1982.

[52] Philip E. Gill, Walter Murray, Michael A. Saunders, and Margaret H. Wright. Shifted barrier methods for linear programming. Report SOL 87-9, Department of Operations Research, Stanford University, Stanford, CA, 1987.

[53] Philip E. Gill and Daniel P. Robinson. A primal-dual augmented Lagrangian. *Comput. Optim. Appl.*, 51:1–25, 2012.

[54] Philip E. Gill and Margaret H. Wright. *Computational Optimization: Nonlinear Programming.* Cambridge University Press, New York, NY, USA, 2023. To be published in 2023.

[55] Philip E. Gill and Minxin Zhang. Numerical results for a projected-search path-following method for nonlinear optimization. Center for Computational Mathematics Report CCoM 22-03, Center for Computational Mathematics, University of California, San Diego, La Jolla, CA, 2022.

[56] Donald Goldfarb, Roman A. Polyak, Katya Scheinberg, and I. Yuzefovich. A modified barrier-augmented Lagrangian method for constrained minimization. *Comput. Optim. Appl.*, 14(1):55–74, 1999.

[57] A. A. Goldstein. Convex programming in Hilbert space. *Bulletin of the American Mathematical Society*, 70(5):709–710, 1964.

[58] Nicholas I. M. Gould, D. Orban, and Philippe L. Toint. CUTEr and SifDec: A constrained and unconstrained testing environment, revisited. *ACM Trans. Math. Software*, 29(4):373–394, 2003.

[59] Nicholas I. M. Gould, Dominique Orban, Annick Sartenaer, and Philippe L. Toint. Superlinear convergence of primal-dual interior point algorithms for nonlinear programming. *SIAM J. Optim.*, 11(4):974–1002, 2001.

[60] William W. Hager. A derivative-based bracketing scheme for univariate minimization and the conjugate gradient method. *Computers Math. Applic.*, 18(9):779–795, 1989.

[61] M. R. Hestenes. Multiplier and gradient methods. *J. Optim. Theory Appl.*, 4:303–320, 1969.

[62] W. Hock and K. Schittkowski. *Test Examples for Nonlinear Programming Codes.* Lecture Notes in Econom. Math. Syst. 187. Springer-Verlag, Berlin, 1981.

[63] C. A. Johnson and Ariela Sofer. A primal-dual method for large-scale image reconstruction in emission tomography. *SIAM J. Optim.*, 11:691–715, 2000/01.

[64] Dongmin Kim, Suvrit Sra, and Inderjit S. Dhillon. Tackling box-constrained optimization via a new projected quasi-Newton approach. *SIAM J. Sci. Comput.*, 32(6):3548–3563, December 2010.

[65] Donald Knuth. *The Art of Computer Programming, 3.* Addison-Wesley Publishing Company, Redwood City, third edition, 1997.

[66] M. W. Leonard. *Reduced Hessian Quasi-Newton Methods for Optimization.* PhD thesis, Department of Mathematics, University of California, San Diego, 1995.

[67] E. S. Levitin and B. T. Polyak. Constrained minimization methods. *U.S.S.R. Comput. Math. and Math. Physics*, 6(5):1–50, 1966.

[68] José Luis Morales and Jorge Nocedal. Remark on "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization". *ACM Trans. Math. Softw.*, 38(1):7:1–7:4, December 2011.

[69] Jorge J. Moré. Recent developments in algorithms and software for trust region methods. In A. Bachem, M. Grötschel, and B. Korte, editors, *Mathematical Programming: the State of the Art (Bonn, 1982)*, pages 258–287. Springer, Berlin, 1983.

[70] Jorge J. Moré and Danny C. Sorensen. Newton's method. In Gene H. Golub, editor, *Studies in Mathematics, Volume 24. MAA Studies in Numerical Analysis*, pages 29–82. Math. Assoc. America, Washington, DC, 1984.

[71] Jorge J. Moré and David J. Thuente. Line search algorithms with guaranteed sufficient decrease. *ACM Trans. Math. Software*, 20(3):286–307, 1994.

[72] Stephen G. Nash, Roman Polyak, and Ariela Sofer. Numerical comparison of barrier and modified-barrier methods for large-scale bound-constrained optimization. In D. W. Hearn and P. M Pardalos, editors, *Large-Scale Optimization: State of the Art*, pages 319–338. Kluwer, Dordrecht, 1994.

[73] Q. Ni and Ya-Xiang Yuan. A subspace limited memory quasi-Newton algorithm for large-scale nonlinear bound constrained optimization. *Math. Comput.*, 66:1509–1520, 10 1997.

[74] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer-Verlag, New York, 1999.

[75] James M. Ortega and Werner C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000. Reprint of the 1970 original.

[76] Todd Plantenga. A trust region method for nonlinear programming based on primal interior-point techniques. *SIAM J. Sci. Comput.*, 20(1):282–305, 1998.

[77] Roman A. Polyak. Modified barrier functions (theory and methods). *Math. Program.*, 54(2, Ser. A):177–222, 1992.

[78] Dulce B. Ponceleón. *Barrier methods for large-scale quadratic programming*. PhD thesis, Department of Computer Science, Stanford University, Stanford, CA, 1990.

[79] Michael J. D. Powell. A method for nonlinear constraints in minimization problems. In Roger Fletcher, editor, *Optimization*, pages 283–298, London and New York, 1969. Academic Press.

[80] David F. Shanno and Robert J. Vanderbei. Interior-point methods for nonconvex nonlinear programming: orderings and higher-order methods. *Math. Program.*, 87(2, Ser. B):303–316, 2000.

[81] D. Siegel. Modifying the BFGS update by a new column scaling technique. *Math. Program.*, 66:45–78, 1994. Ser. A.

[82] André L. Tits, Andreas Wächter, Sasan Bakhtiari, Thomas J. Urban, and Craig T. Lawrence. A primal-dual method for nonlinear programming with strong global and local convergence properties. Technical Report 2002–29, Institute for Systems Research, University of Maryland, College Park, 2002.

[83] Andreas Wächter and Lorenz T. Biegler. Line search filter methods for nonlinear programming: motivation and global convergence. *SIAM J. Optim.*, 16(1):1–31 (electronic), 2005.

[84] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.*, 106(1, Ser. A):25–57, 2006.

[85] J. W. J. Williams. Algorithm 232 - Heapsort. *Communications of the Association for Computing Machinery*, 7:347–348, 1964.

[86] Philip Wolfe. Convergence conditions for ascent methods. *SIAM Rev.*, 11:226–235, 1969.

[87] Philip Wolfe. On the convergence of gradient methods under constraint. *IBM J. Res. Dev.*, 16:407–411, 1972.

[88] Margaret H. Wright. Some properties of the Hessian of the logarithmic barrier function. *Math. Program.*, 67(2, Ser. A):265–295, 1994.

[89] Margaret H. Wright. Ill-conditioning and computational error in interior methods for nonlinear programming. *SIAM J. Optim.*, 9(1):84–111, 1998.

[90] Stephen J. Wright. Stability of linear equations solvers in interior-point methods. *SIAM J. Matrix Anal. Appl.*, 16:1287–1307, 1995.

[91] Stephen J. Wright. *Primal-dual interior-point methods.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[92] Hiroshi Yamashita and Hiroshi Yabe. A primal-dual interior point method for nonlinear optimization: global convergence, convergence rate and numerical performance for large scale problems. In *Parametric Optimization and Related Topics, V (Tokyo, 1997)*, pages 213–250. Lang, Frankfurt am Main, 2000.

[93] Yin Zhang, Richard Tapia, and Florian Potra. On the superlinear convergence of interior-point algorithms for a general class of problems. *SIAM J. Optim.*, 3(2):413–422, 1993.

[94] Yin Zhang and Richard A. Tapia. A superlinearly convergent polynomial primal-dual interior-point algorithm for linear programming. *SIAM J. Optim.*, 3(1):118–133, 1993.

[95] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans. Math. Software*, 23(4):550–560, 1997.