
THE UNIVERSITY *of* EDINBURGH

THE UNIVERSITY *of* EDINBURGH

# Active-set Prediction for Interior Point Methods

*Yiming Yan*

Doctor of Philosophy
University of Edinburgh
November 22, 2014

# Declaration

I declare that this thesis was composed by myself and that the work contained therein is my own, except where explicitly stated otherwise in the text.

(*Yiming Yan*)

November 22, 2014

*To my mother and father, who have supported me with love and trust through my life. And to my wife, whose love and confidence is a constant source of inspiration and encouragement.*

# Abstract

This research studies how to efficiently predict optimal active constraints of an inequality constrained optimization problem, in the context of Interior Point Methods (IPMs). We propose a framework based on shifting/perturbing the inequality constraints of the problem.

Despite being a class of powerful tools for solving Linear Programming (LP) problems, IPMs are well-known to encounter difficulties with active-set prediction due essentially to their construction. When applied to an inequality constrained optimization problem, IPMs generate iterates that belong to the interior of the set determined by the constraints, thus avoiding/ignoring the combinatorial aspect of the solution. This comes at the cost of difficulty in predicting the optimal active constraints that would enable termination, as well as increasing ill-conditioning of the solution process. We show that, existing techniques for active-set prediction, however, suffer from difficulties in making an accurate prediction at the early stage of the iterative process of IPMs; when these techniques are ready to yield an accurate prediction towards the end of a run, as the iterates approach the solution set, the IPMs have to solve increasingly ill-conditioned and hence difficult, subproblems.

To address this challenging question, we propose the use of controlled perturbations. Namely, in the context of LP problems, we consider perturbing the inequality constraints (by a small amount) so as to enlarge the feasible set. We show that if the perturbations are chosen judiciously, the solution of the original problem lies on or close to the central path of the perturbed problem. We solve the resulting perturbed problem(s) using a path-following IPM while predicting on the way the active set of the original LP problem; we find that our approach is able to accurately predict the optimal active set of the original problem before the duality gap for the perturbed problem gets too small. Furthermore, depending on problem conditioning, this prediction can happen sooner than predicting the active set for the perturbed problem or for the original one if no perturbations are used. Proof-of-concept algorithms are presented and encouraging preliminary numerical experience is also reported when comparing activity prediction for the perturbed and unperturbed problem formulations.

We also extend the idea of using controlled perturbations to enhance the capabilities of optimal active-set prediction for IPMs for convex Quadratic Programming (QP) problems. QP problems share many properties of LP, and based on these properties, some results require more care; furthermore, encouraging preliminary numerical experience is also presented for the QP case.

# Acknowledgements

First and foremost I would like to express my deep sense of gratitude to my supervisor, Dr. Coralia Cartis. She has been a tremendous mentor for me. I would like to thank her for her patient guidance, enthusiastic encouragement and useful critiques of my research work. Her advice on my research as well as my career have been priceless. Without her invaluable guidance and support, my completion of this thesis would not have been accomplished.

I would also like to thank Prof. Nick Gould for useful discussions and insights regarding my research project.

I am also very grateful to Prof. Ken McKinnon for his kind encouragement and support during the last year of my Ph.D. study. His advices on future career are of great importance to me.

I am obliged to staff members of the School of Mathematics at the University of Edinburgh, for the valuable information and help provided by them.

I also would like to take this opportunity to acknowledge the financial support I received during my Ph.D. research. I am grateful to the University of Edinburgh for granting me the Principal's Career Development Ph.D. Scholarship and the Edinburgh Global Research Scholarship.

It has been a good time sharing Room 4602 with Waqquas Bukhsh and Feng Qiang. They have been great office mates and friends. And also I would like to thank my friends for the time spent together. Special thanks go to Sarah Farid, Andrea Meireles, Dulip Piyaratne and Jinhan Xie.

It has been fun as a committee member of the Edinburgh SIAM Student Chapter. I am indebted to all other committee members for their support and trust. Especially I thank Pablo González Brevis for guiding me through all the work and Martin Takáč for his contribution to the student chapter.

A special thanks to my loved ones, my parents and my wife, who have supported me throughout the entire process, both by keeping me harmonious and helping me putting pieces together. I will be grateful forever for their love and support as well as their patience for listening to me when I was stressed and upset with myself.

# Contents

# List of Tables

# List of Figures

# List of Algorithms

# 1 Introduction and Motivation

In this chapter, we present a brief introduction to Linear Programming (LP) and Quadratic Programming (QP), as well as to some of the most important state-of-art algorithms for solving this class of problems. We also present a comprehensive overview of existing active-set prediction techniques. Next, we state our motivation and give an outline of this thesis' contents.

## 1.1 Linear Programming

Optimisation is "built into nature" [62], as described by Leonhard Euler (1744):

> *Nothing in the world takes place without optimisation, and there is no doubt that all aspects of the world that have a rational basis can be explained by optimisation methods.*

From the origin of our species, human beings desire to make effective decisions. Investors attempt to gain a good profit while avoiding high risk. Engineers intend to design more efficient and less expensive systems. Manufacturers aim to maximally utilise their production lines. Pharmacists strive to find the best medicine formulation to battle disease. Car drivers seek to find the shortest routine to save time and fuel. Nature also optimises. Physical systems tend to a position with minimal potential energy. Lights travel through the path that requires least travel time. Optimisation is a procedure that involves describing such problems mathematically and finding the 'best' solution.

Linear Programming (LP) is regarded as one of the most widely used and well-established optimisation models for real world problems. It was initially developed for the needs to solve complex military planning problems in World War II. The development in this area blossomed after the war, due partly to the realisation of its valuable uses in industry. Applications of LP models arise in many different scenarios such as management, investment, transportation, scheduling, telecommunication, etc. The world, however, is full of complex problems that are not so simple as expressed by the linear relations that underlie the LP model. More sophisticated nonlinear models

are studied to better reflect the features of such problems. This does not cast a shadow over LP models as many nonlinear relations can be approximated by a set of linear equalities/inequalities. For descriptions of the history of LP, see [27, Chpater 2] and *Notes on polyhedra, linear inequalities, and linear programming* in [118].

LP is an optimisation problem for which we intend to minimise (or maximise) an *objective function* consisting of a linear combination of decision *variables*, under a set of linear *constraints*, both equalities and inequalities. A particular formulation of LP can be expressed as

$$\min_{x \in \mathbb{R}^n} c^T x \quad \text{subject to} \quad Ax = b, \quad x \geq 0, \tag{1.1}$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $c \in \mathbb{R}^n$. Geometrically, the *feasible region* $\mathcal{P}^x = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\}$ is a polyhedron which is the intersection of linear constraints in the space of the variables. To solve an LP problem is to find a point in $\mathcal{P}^x$ that minimises or maximises the objective function $c^T x$. The optimal value of the objective function can generally be achieved at a vertex of the polyhedron $\mathcal{P}^x$ [105].

If we refer to (1.1) as the *primal problem*, there always exists an associated *dual problem*, which has the same optimal objective value as (1.1) but interprets the same set of data from a different point of view. The dual problem of (1.1) can be expressed as

$$\max_{(y,s) \in \mathbb{R}^{m \times n}} b^T y \quad \text{subject to} \quad A^T y + s = c, \quad y \text{ free}, \ s \geq 0. \tag{1.2}$$

The most famous and powerful classes of methods to solve LP problems are the *simplex method(s)* and *interior point methods*, which are introduced in the following two sections.

## 1.2 Simplex method

The simplex method for linear programming, introduced by George B. Dantzig [27] in the late 1940s, is regarded as one of the major breakthrough in optimisation of the 20$^{\text{th}}$ century. For several decades, this was the main practical method to solve LP [54]. The simplex method searches along the boundary of the polyhedron $\mathcal{P}^x$ defined by the constraints. It moves from one vertex to an adjacent one with a better value of the objective function, thus exploring the combinatorial structure of the problem; see Figure 1.1 for an illustration.

A feasible vertex, namely a vertex of the feasible polyhedron $\mathcal{P}^x$, is also called a *basic feasible point*. A basic feasible point becomes a *vertex solution* if the optimal value is obtained at the given point. Due to the way in which the simplex method operates, it is also considered as an *active-set method* [39].

Although in the worst case the possible number of steps of the simplex method to find the solution may depend exponentially on the problem dimensions (Klee & Minty [72]), practical implementations are efficient and reliable [65]. It has been im-

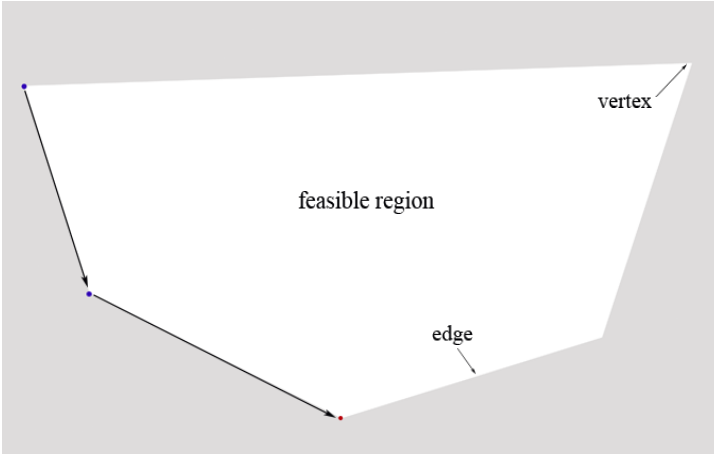



Figure 1.1: An illustration of the simplex method

plemented in modern optimisation solvers such as CPLEX [68]. Other simplex-method-based open source software includes LP_SOLVE [14], GLPK [2], CLP [1], etc.

## 1.3 Interior point methods

Having its origin with Dikin's work [31] and Khachiyan's [71] and then refined by Karmarkar [70], interior point methods (IPMs) are a class of very efficient tools to solve LP problems, especially for problems whose dimensions are very large. Contrary to the simplex method, IPMs reach an optimal solution by travelling through the relative interior of the feasible polyhedron $\mathcal{P}^x$ without visiting the possibly-many feasible vertices, approximately following the so-called *central path*; see Figure 1.2 for an illustration.

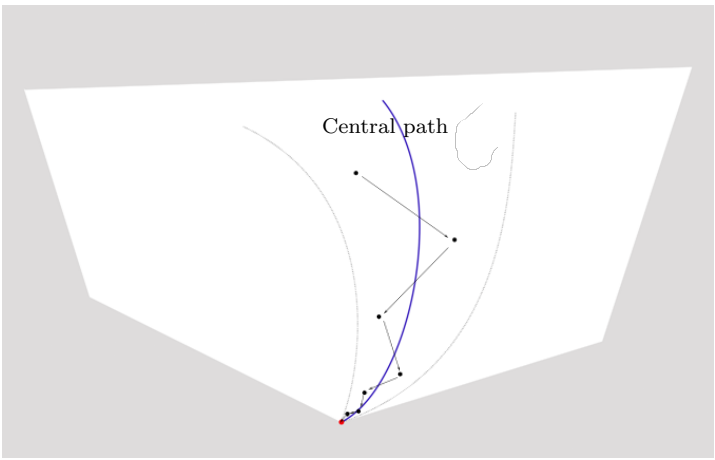



Figure 1.2: An illustration of a primal-dual path-following IPM

Following the publication of Karmarkar's method, Gill et al. [45] revealed the equivalence between Karmarkar's method and the projected Newton barrier method using *logarithmic barrier functions* [37]. The logarithmic barrier function was originally proposed by Frisch [43] in 1955 and extensively studied by Fiacco and McCormick [37] for

nonlinear programming. The promising implementation in [45] started a new era for IPMs, which led to the development of the *path-following algorithms*.

In general, path-following IPMs 'remove' the inequality constraints by appending them to the objective with the help of the logarithmic barrier function; then they approximately solve the resulting subproblem. When applying the method to (1.1), we replace $x \geq 0$ with the logarithmic barrier function $-\mu \sum_{j=1}^{n} \log x_j$, and have the barrier subproblem

$$\min_{x \in \mathbb{R}^n} c^T x - \mu \sum_{j=1}^{n} \log x_j \quad \text{subject to} \quad Ax = b, \tag{1.3}$$

where the positive scalar $\mu$ is known as the *barrier parameter*. This parameter controls the relation between the barrier term and the original LP objective. A large value of $\mu$ corresponds to preventing the points $x$ from approaching the boundaries of the feasible region $\mathcal{P}^x$ of (1.1) (as the barrier term blows up at a boundary); a smaller value of $\mu$ reduces the influence of the barrier term and more attention is paid to the original LP objective.

Under certain conditions (see Theorem 2.6), problem (1.3) has a unique solution $x(\mu)$, for every $\mu > 0$ [45]. If we continuously decrease the value of $\mu$ from a large value to zero, the corresponding solutions $x(\mu)$ will form a path $\mathcal{C}_P$, which moves towards an optimal solution (the *analytic centre* of the optimal solution set) of (1.1). This path is known as the *central path*. The central path $\mathcal{C}_P$ of (1.1) is known as the *primal central path*, since it only concerns the *primal problem*. We can also derive a similar central path for the dual problem (1.2), which converges to the analytic centre of the dual solution set. The variant of IPMs that considers the primal-dual central path is called the *primal-dual path-following* IPM*s*, which is considered as the most successful class of IPMs; see Section 2.2 for in-depth descriptions of this class of methods, including a general algorithmic framework (Algorithm 2.1). There exist many theoretical and practical algorithms that fit exactly or approximately into the framework of the primal-dual path-following IPMs such as short step [75, 100], long step [76], predictor-corrector [99], infeasible [74], and Mehrotra predictor-corrector [93, 83] IPMs.

Newton's method [30] is used to compute an approximate solution of the barrier subproblem (1.3) in its primal/dual or primal-dual form. One or more Newton steps are computed for a fixed value of the barrier parameter $\mu$. This procedure is repeated for a sequence of decreasing values of $\mu$, until $\mu$ is small enough to ensure proximity to the solution of the original LP problem. During each iteration, the Newton search direction is defined by a certain linear system at the current iterate, some control parameters and $\mu$. Solving such a system is the main computational cost of each IPM iteration; see Section 2.2 for the formulation of the Newton direction and how to calculate it.

The coefficient matrix of the linear system mentioned above becomes more and more ill-conditioned as the iterates approach the solution set. This may lead to unacceptable difficulty in meeting convergence criteria and undesirably large residuals. The

effects of ill-conditioning for IPMs, however, can be relatively benign, mainly due to the development of modern linear algebra techniques [132, 131, 133]. Another way to avoid ill-conditioning is to terminate the interior point iterative process early on and crossover to simplex-like methods, before $\mu$ gets too small or the iterates get too close to the boundaries; see the corresponding descriptions of 'crossover' in Section 1.4.

**Computational complexity of linear programming.** Computational complexity has played an important role in the history of linear programming, leading to the research of algorithms with better — namely polynomial — complexity than the simplex method and thus the birth of interior point methods. The computational complexity of a problem instance evaluates the total amount of computational cost (running time, memory) required to solve the problem instance "in some measure of the problem data" [135]. There are different models of analysing computational complexity, such as real-number model [20] and rational-number (Turing) model, with the latter being most commonly used in interior-point methods literature; see [135, Chapter 3] for details of different computational models. In the rational-number model, the data $(A, b, c)$ of (1.1) is assumed to be rational and thus can be stored exactly in a computer. Let $L$ denote the total length of the data string required to store all problem data. Then the complexity results are often an expression of $L$ and $n$, where $n$ is referred to as the *problem dimension*, namely the number of variables for LP[1]. We restrict our discussion in the following paragraphs to the rational-number model.

The worst-case complexity measures the 'worst-case behaviour' of a given algorithm, namely it gives an upper bound on the computational cost required by the algorithm to solve any problem instance from a chosen class [135]. Thus an algorithm is said to have *polynomial worst-case complexity* if the computational complexity of solving any problem instance from a chosen class is bounded above by a polynomial in data string storing all data and the problem dimension, namely $(L, n)$.

The ellipsoid algorithm developed by Khachiyan [71] was the first polynomial algorithm for linear programming, which requires at most $\mathcal{O}(n^2L)$ iterations to find an optimal solution and at most $\mathcal{O}(n^4L)$ arithmetic operations. This worst-case bound, however, is achieved on most problems and in practice its performance is generally worse than the simplex method [135, Page 57]. Karmarkar [70] proposed a polynomial algorithm with an iteration bound of $\mathcal{O}(nL)$ and lowered the overall worst-case computational complexity to $\mathcal{O}(n^{3.5}L)$ arithmetic operations. Renegar [112] then improved this worst-case iteration bound to $\mathcal{O}(\sqrt{n}L)$, which remains the best known worst-case iteration complexity bound for IPMs for LP. After that, Anstreicher [6] lowered the overall worst-case complexity to $\mathcal{O}(\frac{n^3}{\ln n}L)$ arithmetic operations. Please refer to [135, Chapter 3] or [6] for details of the complexity results for IPMs for LP.

[1]Some complexity results also include $m$, the number of rows of $A$. If we assume $A$ has full row rank and so $m \leq n$, the complexity bounds always hold if we replace $m$ with $n$ [135, Chapter 3].

**Implementation and software.** The most successful commercial or public IPM codes are generally based on the Mehrotra Predictor-Corrector (MPC) algorithm [135, Chapter 10], which has been shown numerically to be much faster than other IPM approaches especially for large-scale problems, although no global convergence or polynomial complexity results are known for the variant implemented in state-of-art codes. Popular IPM implementations include the barrier solver of CPLEX (CPLEX Optimization Inc.), MOSEK (Erling D. Andersen et al.), HOPDM (Gondzio et al.), PCx (Czyzyk et al.), LOQO (Vanderbei et al.), LIPSOL (Zhang et al.), etc.

**Large-scale optimisation.** Interior point methods are in general more efficient than the simplex method when solving large-scale problems [18, 54, 82, 83, 93]. Successful application of IPMs to solving large-scale problems has occurred in different scenarios, for instance, Gondzio and Grothey [57] solved a nonlinear portfolio optimisation problem with $10^9$ variables and Koh et al. [73] used a variant of IPMs to solve a large machine learning problem with millions of features and examples in under an hour on a PC. Other applications of IPMs to large-scale problems can also be found in [97].

**Surveys.** For a history of interior point methods, please refer to the survey by Potra and Wright [111] and the more recent one by Gondzio [54]. For more technical details on IPMs for linear programming, see Wright's comprehensive textbook [135] and Nocedal and Wright [105, Chapter 14]. A description of IPMs for general convex optimization problems can be found in [21, Chapter 11] and in Renegar's dedicated monograph [113].

## 1.4 Active-set prediction

Consider an inequality-constrained optimisation problem, which minimises (or maximises) the objective function over the feasible region composed of points satisfying the constraints. An *active constraint* is an inequality constraint that holds as equality at a feasible point. *Optimal active-set prediction* — namely, identifying the active inequality constraints at the solution of a constrained optimisation problem — plays an important role in the optimisation process by removing the difficult combinatorial aspect of the problem and reducing it to an equality-constrained one that is in general easier to solve [35]. Consider LP problems as an example: if we know the correct active constraints at the optimal solution, we are able to locate the optimal vertex directly by solving a simple linear system of equations without having to consider the combinatorially large number of vertices of the polyhedron; see an illustration in Figure 1.3. A *strongly active constraint* is an active constraint whose corresponding Lagrange multiplier[2] is strictly positive; for details, see [105, Definition 12.8]. The set of all strongly active constraints is called the *strongly active set*. It is straightforward to see that the strongly active set is a subset of the active set.

[2]For an inequality constraint $x_i \geq 0$ in (1.1), its Lagrange multiplier is the dual variable $s_i$ in (1.2), $i = 1, \ldots, n$; for the definition of Lagrange multiplier, please refer to [105, Chapter 12].

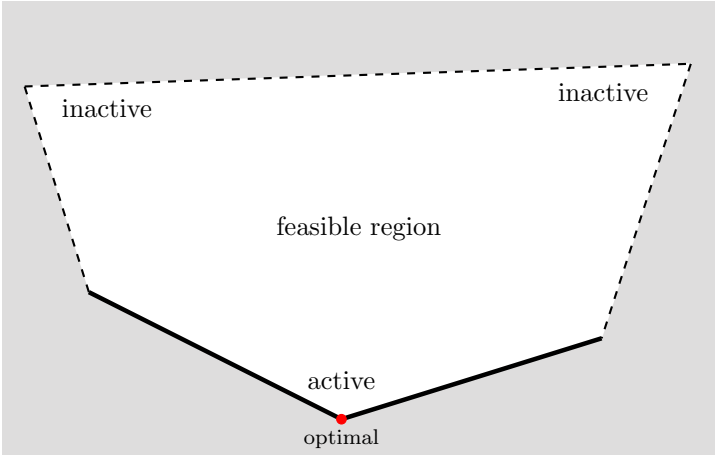



Figure 1.3: An illustration of active constraints in the context of linear programming

Active-set prediction has been part of optimisation techniques and literature for decades [39] and it usually takes place while running an iterative algorithm. Existing active-set prediction techniques have been developed mostly in the context of Nonlinear Programming (NLP), while some of them have been proposed specifically for IPMs.

**Active-set prediction for NLP.** Fletcher [38] proposed an active-set algorithm for solving quadratic programming problem, an essential part of which concerns the strategy of adding and/or deleting predicted active constraints. In this approach, no more than one constraint should be exchanged from one iteration to the next. On the other hand, Goldfarb [49] presented a method where more than one constraint could be deleted from the predicted active set. Lenard [77] numerically compared different procedures for determining active constraints at each iteration in the context of a quasi-Newton projection method for nonlinear problem with linear constraints and concluded that keeping the predicted active set as small as possible considerably saves computing time when the optimal solution is not a vertex; otherwise, the active set should be as large as possible. Bertsekas [16] proposed a two-metric algorithm for solving an NLP subject to only simple bound constraints, with a strategy of estimating the active bounds at a local solution. When the iterates fall into a neighbourhood of a local solution, strongly active constraints[3] at this solution can be predicted. This is also proved by Lescrenier [78] for a trust region method. Different methods have been proposed for identifying/predicting the optimal active set for the more general case when the feasible points form a convex set. In this context, optimal active-set prediction corresponds to identifying the *optimal face* upon which the objective function attains its (local) optimal value; see [22] for a technique for predicting the "quasi-polyhedral" faces ([22, Definition 2.5]) near a local solution, and [134] for identifying the "class-$\mathcal{C}^p$ identifiable surface" under a nondegeneracy assumption.

In addition to the above methods, Facchinei, Fischer, and Kanzow [35] describe

[3] Please refer to the description of the strongly active constraints and strongly active set on page 6.

the use of a class of special functions, the identification functions, which provide an estimate of the distance from the current iterate to the solution set. Based on this estimate, threshold tests are proposed to predict the active and strongly active sets[3] in a neighbourhood of a local solution. They state that the formulation of an identification function is "the crucial point in the identification of active constraints". We discuss this technique in detail and propose to apply it to LP in Section 3.1. Oberlin and Wright [106] define a new identification function that requires a linear programming problem to be solved in order to predict the active set at a local optimum. For more results based on [35, 106], see, for example, [80] for an extension to linearly constrained minimisation without derivatives, [79] for the composite nonsmooth minimisation problem, [66] for constrained minimax problems, and [121] for large scale bound constrained optimization.

Similar active-set prediction techniques can be extended to the monotone Linear Complementarity Problem (LCP) and the monotone Nonlinear Complementarity Problem (NCP); see, for example, the paper of Monteiro and Wright [102] for the former and Yamashita, Dan, and Fukushima [136] for the latter.

**Active-set prediction for IPMs for LP.** Various ways have been devised for IPMs to predict the optimal active set during their run. The simplest one is to conduct a threshold test with a fixed constant [45, 69, 89], namely, split the variables into active or inactive based on whether they are less than a user-defined small value. The most well-known class of active-set prediction techniques for IPMs is *indicators* [32] which form functions of iterates and identify the optimal active-set based on whether the values of these functions are less than a threshold; we discuss the formulations of two state-of-art indicators [32] in Section 3.2. Mehrotra [92] suggests determining the active set by a simple comparison of the relative increments of primal and dual iterates, and Mehrotra and Ye [94] propose a strategy to identify the active set by comparing the primal variables with the dual slacks; see [129] for a review of active-set prediction techniques for IPMs for LP.

**Applications of active-set prediction techniques.** Here we give some examples of applications of active-set prediction techniques in different scenarios.

- **Warmstarting**. Warmstarting is a technique that uses the information obtained from solving an initial LP problem to accelerate the re-optimisation of one or more closely related problems with only minor changes from the original problem structure. If the active set of the closely related problem can be identified from the optimal or nearly-optimal solution of the original problem, it may greatly enhance the efficiency of the warmstarting process and hence the solution time for the perturbed problem. Related active-set prediction techniques have been developed in this context; see for example, the surveys [34, 120]. We briefly review relevant contributions here. One of the main warmstarting strategies focuses on

the 'iterates', namely it manipulates the (IPM-computed) near optimal or optimal iterates of the initial problem to obtain a primal-dual feasible and well-centred point for the perturbed problems, see for example, [53, 60, 139, 58, 120]. Another category of approaches works on the 'problem formulation', namely modify the problem formulation by relaxing the nonnegativity constraints in the form of shifted logarithmic barrier variables, which has some similarity to our approach. Earlier works in this framework include Freund [40, 42, 41], Mitchell [96] and Polyak [110] with promising theoretical properties. Freund's papers [40, 42] only relax the nonnegativity constraints for the primal problem and his paper [41] relaxes the equality constraints of the primal and dual problem. In [110], Polyak applies modified barrier functions to derive IPMs, but still only relaxes the primal nonnegative variables. Mitchell's paper [96] applies Freund's method in [40] to column generation. These methods only loosely relate to our approach. More relevant warmstarting approaches to our idea are Benson and Shanno [11] and Engau, Anjos and Vannelli [33, 34]. The former proposes a primal-dual penalty strategy relaxing the nonnegativity constraints for both primal and dual decision variables and then penalising the relaxation variables in the objective; the latter applies a simplified primal-dual slack approach introducing slack variables for nonnegative constraints and penalising the slack variables in the objective. For the difference between these approaches and our method, please refer to Section 1.6.

- **Crossover to simplex and basis recovery**. When the given LP problem has multiple solutions, IPMs find a solution in the relative interior of the solution set instead of a vertex solution. Since it is difficult to perform post-optimality analysis from such an 'interior' solution [19], obtaining a basic optimal solution (i.e, a vertex solution) is essential for some applications, such as determining 'shadow prices' when allocating resources [67, Section 4.7]. Thus it makes sense to convert the IPM solution to a basic feasible solution and then switch ('crossover') to the simplex method from this basic feasible solution. Such a procedure is known as crossover [28, 114, 135]. A good prediction of the optimal active set is essential for crossover since it will provide a good approximation of the basic and nonbasic partition and thus help generate a basic feasible point that is close to the vertex solution. Similar ideas also appear in the literatures of 'basis recovery'. Tapia and Zhang [122] proposed an indicator, which uses the diagonal information of the matrix from normal equations (for details, see (2.16)), to predict the optimal basis during the run of IPMs. Other than directly predicting basic and nonbasic variables, there are also methods that apply simplex-like pivoting strategies to recover the basis from an (almost) exact or approximate optimal solution generated by IPMs; please refer to [91, 19, 5]. For a general description of crossover, see also [135, Chapter 7]. The functionality of crossover has been implemented in many optimisation routines of different pieces of software, such as CPLEX [68], LINGO [81], and SAS [115], which reveals its importance in real world applications.

- **Finite terminations for IPMs**. Active-set prediction is closely related to the so-called *finite termination*, which is a technique that calculates an exact solution in a finite number of steps by projecting the current solution estimate onto the solution set. This is especially useful for IPMs because IPMs only converge to the solution set asymptotically [137]. A good active prediction is crucial to ensure good performance of finite termination procedures.

- **Constraint reduction for IPMs**. Tits, Absil and Woessner [125] propose to use the dual active set to form a reduced version of normal equations (see (2.16)), so as to reduce the computational cost when calculating the Newton directions. Winternitz et al. [130] present a result based on [125] for Mehrotra's Predictor-Corrector (MPC) Algorithm and give the global convergence under significantly milder assumptions than [125]. Encouraging numerical results are presented.

- **Multicommodity flow problems**. Babonneau, Merle and Vial [8] propose an active-set strategy, a threshold test on the capacity usage of an arc, in order to reduce the problem dimension, when solving large-scale linear multicommodity flow problems. Babonneau and Vial [9] extend this strategy to nonlinear constrained multicommodity flow problems by applying it to arcs where the cost function can be approximated by a linear function.

- **Column generation**. Gondzio, González-Brevis and Munari [56] incorporate a similar strategy as [8] in their implementation of a primal-dual column generation method where the algorithm starts with optimising a subproblem with a small set of constraints. They improve the cost by adding new potentially active constraints at each iteration. For more literature in this area, please refer for example to [95, 48, 98, 7].

- **Machine learning**. De Leone and Lazzari [29] also employ the idea of the identification function to predict the active constraints of a specific QP problem arising in the area of Support Vector Machines (SVMs).

## 1.5 Quadratic programming

A Quadratic Programming (QP) problem minimises or maximises a quadratic function subject to linear constraints. QP models are also widely used for solving real world problems, such as Markowitz mean-variance portfolio optimisation problem [86] in finance, demand-supply response [88] in economics and electrical energy production [107, 104] in engineering. Without loss of generality, a particular formulation of QP can be expressed as

$$\min_{x\in\mathbb{R}^n} \quad c^T x + \frac{1}{2} x^T H x \quad \text{subject to} \quad Ax = b, \quad x \geq 0, \tag{1.4}$$

where $H \in \mathbb{R}^{n\times n}$ is symmetric, $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, and $c \in \mathbb{R}^n$. If the Hessian matrix $H$ is positive semidefinite, then (1.4) is a *convex* QP problem. In this thesis,

we always consider convex QP problems. A local minimum of a convex QP is always a global minimum, namely the general first-order necessary conditions given to QP are also sufficient for a global minimum when $H$ is positive definite; otherwise, we can only say that they are necessary. In addition, if the QP problem is convex, it is solvable in polynomial time; otherwise it is NP-hard [109]. The feasible region of the QP problem (1.4), similarly to that of the LP problem (1.1), is also a polyhedron, but the optimal solution of (1.4) may be found anywhere within the polyhedron or on its surface. See Figure 1.4 for an illustration.

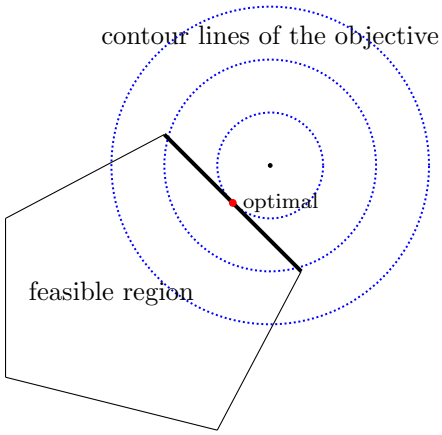



Figure 1.4: An illustration of the optimal solution of a two-dimensional QP problem

Active-set methods are a main class of methods for solving QP, which move from a feasible point towards a solution along the edges and faces of the feasible set. Each iteration involves solving an equality-constrained QP. Active-set methods for QP differ from the simplex method for LP in that neither the iterates nor the optimal solution are required to be vertices of the feasible polyhedron. For details, please refer to a classic book of Fletcher [39] or a more recent survey by Gill and Wong [47].

The interior point methods for QP are general extensions from those for LP with the coefficient matrix $H$ of the quadratic terms taken into consideration when solving for the Newton directions [54]. The presence of $H$, however, makes the Newton system much more costly to solve than that arising in linear programming [105]. Exploiting the structure of the coefficient matrix of the Newton system [59], or, alternatively, employing an appropriate preconditioner for an iterative solver (such as projected CG method) [55] is often needed.

IPMs for LP converge to a so-called strictly complementary solution (which always exists for LP; see Section 2.1) which leads to a unique optimal active and inactive partition of the constraints. IPMs for QP, however, may not find such a solution, due essentially to the fact that such a solution may not exist for QP [12, 13]. This leads to the analysis of the 'tripartition' (Section 2.3) instead of the optimal active and inactive partition, which is the main reason why we perform separate analyses of active-set predictions for LP and QP in this thesis.

Existing active-set solvers for QP include QPOPT [46] and QPA [61]; for interior

point solvers, please refer to QHOPDM [3] (QP version of HOPDM [51]) and LOQO [127] for example.

## 1.6 Motivation and outline of thesis

In this thesis, we focus on developing active-set prediction/identification techniques for IPMs in the context of linear and quadratic programming.

Active-set prediction is trivial for simplex method, since it moves from one vertex to an adjacent one, and on every iteration it has a working active set. IPMs, however, are well-known to encounter difficulties with active-set prediction due essentially to their construction. They generate iterates that progress towards the solution set through the (relative) interior of the feasible set, and thus avoid visiting possibly-many feasible vertices (see Figure 1.2 for an illustration). This may also prevent IPMs from getting accurate information about the optimal active set early enough during their running. When this information is more readily predictable/available towards the end of a run, as the iterates approach the solution set, the algorithm has to solve increasingly ill-conditioned and hence difficult, subproblems.

Finding ways to improve (even just partial) active-set prediction for IPMs could thus be beneficial as it would allow earlier termination of an otherwise ill-conditioned and computationally expensive process, by say, projecting onto the solution set (as in finite termination [137]). It can also help with reducing the problem size or with obtaining a vertex solution at the cost of just a few additional (and less expensive) simplex method iterations (see Section 1.4 earlier).

Although active-set prediction techniques for IPMs have existed for over a decade, their performances could be improved due to the difficulties in making an accurate prediction at the early stage of the iterative process of IPMs. In the case of indicators [32] for example, to get a good prediction, the iterates still need to be close to optimality (small duality gap). For instance, in Table 8.2 in [32], at the third from the last iteration, 3 out of the 6 problems predict only a very small portion of the active constraints (less than 15%) using Tapia indicators. Similar behaviour of indicators is observed in our numerical tests; see Section 3.4 for details.

In this thesis, our main aim is to develop active-set prediction techniques that can predict a large portion of the optimal active set before reaching the very end of the iterative process (before the duality gap is too small), in the hope of decreasing the level of ill-conditioning.

To this end, we propose the use of *controlled perturbations* [24] for active-set prediction for IPMs. Namely, we perturb the inequality constraints of the LP problem (by a small amount) so as to enlarge the feasible set of the problem, then solve the resulting perturbed problem(s) using a path-following IPM while predicting on the way the active set of the original LP problem. As Figure 1.5 illustrates, provided the perturbations are chosen judiciously, the central path of the perturbed problem may pass close to the optimal solution of the original LP problem when the barrier parameter for the perturbed

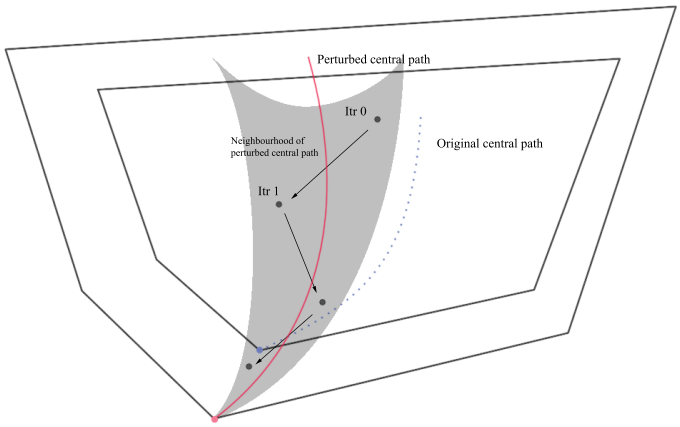



Figure 1.5: Enlarge the feasible set and predict the original active set

problem is 'not too small'. Thus we expect that while still 'far' from optimality for the perturbed problem, some IPM iterates for the perturbed problem would nonetheless be close to optimality for the original LP problem (such as the third and fourth iterate in Figure 1.5) and would provide a good prediction of the original optimal active set. As it may happen that the chosen perturbations are 'too large' or not sufficiently effective for active-set prediction, we allow them to shrink after some IPM iterations so that the resulting perturbed feasible set is smaller but still contains the feasible set of the original LP.

**Relevant existing literature.** Using controlled perturbations was first introduced by Cartis and Gould [24] for finding well-centred points in Phase I of IPMs, a different focus and approach than here.

Since we employ perturbed problems, albeit artificially, our proposal may be reminded of warmstarting techniques for IPMs and the related active-set prediction techniques that have been developed in that context; see relevant literature reviews on page 8. More relevant and closer in spirit to our approach here is [11], where Benson and Shanno propose a primal-dual penalty strategy relaxing the nonnegativity constraints for both primal and dual decision variables and then penalising the relaxation variables in the objective; encouraging numerical results are also reported. Engau, Anjos and Vannelli [33, 34] apply a simplified primal-dual slack approach: instead of shifting the bounds and penalising the relaxation variables, slack variables for nonnegative constraints are introduced and penalised in the objective. One of the main differences between the above techniques and our approach is that we consider perturbations as parameters, not variables that are updated in the run of the IPM; furthermore, our focus is different from that of warmstarting as we specifically aim to predict the active set of the original optimisation problem by using these 'fake' perturbations.

Another technique, regularisation for interior point methods [116, 117, 4], is also 'loosely' related to our approach. In order to improve the conditioning of the coeffi-

cient matrix arising in calculating Newton directions, regularisation terms are added to the objective function (and sometimes the equality constraints [116, 117]). With extra terms present on the diagonal, the resulting augmented system of the regularised problem can be considered as a 'perturbed' version of the original one. The regularisation method, however, does not allow negative components of the primal and dual variables (they do not perturbed the inequality constraints) and is not developed for active set prediction.

**Contributions.** When considering the proposed perturbed problems (see ($PD_\lambda$) on page 46), we show that under certain conditions, the optimal solution of the original problems lies on or in a neighbourhood of the central path of the perturbed problems. Furthermore, we derive that under certain non-degeneracy assumptions, the perturbed problem has the same active set as the original one. We also prove that under certain conditions that do not necessarily require problem nondegeneracy, our predicted active sets bound well the optimal active set of the original (unperturbed) LP, and exactly predict it under a certain nondegeneracy assumption (but without requiring that the perturbed active set coincides with the original one). We also find conditions on problem conditioning that ensure that our prediction of the optimal active set of the original LP can happen sooner than the prediction of the optimal active set of the perturbed problems (so that our approach may not need to solve the perturbed problems to high accuracy). Similarly, we characterise the situations when our approach allows an earlier prediction of the original active set as compared to the case when we solve and predict the original LP directly.

In the preliminary numerical tests for LP, we carry out two type of tests, one comparing the accuracy of the predicted active sets and the other one exploring the case of crossover to simplex method. When verifying the accuracy of our active-set predictions using certain correctness comparison ratios, we observe that when using perturbations, the precision of our predictions is generally higher than that when we do not use perturbations. When crossing over to simplex method, we test the efficiency of our active-set predictions by comparing the number of simplex iterations needed to solve the original problem to optimality, after some initial IPM iterations. We find that when using perturbations for the IPM iterations, we can save (on average) over 30% simplex iterations compared to the case of not using any perturbations before crossover to simplex.

We then extend the prediction results to QP and also derive theorems on predicting the optimal tripartition of a QP problem without the strictly complementary assumption. Although our prototyped algorithm in MATLAB is not optimized or efficient enough and so we can not test large scale problems, the tests on random test problems and the small sized QP problems from Netlib and Maros and Meszaros' convex QP test set do show some promising performance.

**Outline of thesis.** In Chapter 2, we review the theoretical fundamentals needed for the results of this thesis. We first introduce the terms related to LP, including the standard form, solution set, degeneracy, multiplicity, etc (Section 2.1). Section 2.2 is concerned with some basic concepts of the primal-dual path-following IPMs for LP, such as the central path and its neighbourhoods, general algorithmic framework and implementations. In Section 2.3, the basics of QP are introduced. Section 2.4 focuses on deriving error bounds for both linear and quadratic programming problems, which are extensively used in this thesis.

Chapter 3 introduces the details of existing techniques for active-set predictions for IPMs, namely the identification function, indicators and simple cut-off. Numerical experiments are conducted to compare the performance (accuracy, etc.) of these techniques. Furthermore, we investigate the limitations of current techniques.

In Chapter 4, we investigate the use of controlled perturbations for active-set prediction for IPMs for LP, after introducing them and the associated primal-dual perturbed LP problems (Section 4.1). We then show the relations between the solution of the original LP problem and the central path of the perturbed problems (Section 4.2.1), as well as the relation between the optimal active set of the perturbed problems and the original problems (Section 4.2.2).

In Chapter 5, we present our main theoretical results for the active-set prediction for LP. After introducing some useful preliminary results (Section 5.1), we present our main prediction results for LP in Section 5.2. Comparisons between the perturbed and unperturbed active-set predictions are shown in Section 5.3.

In Chapter 6, we present the perturbed algorithm framework for LP (Section 6.1). Then we introduce the test problems and show some useful observations related to the perturbed problems (Section 6.2). In our preliminary numerical experiments for LP, we conduct tests with the simple cut-off (Sections 6.3) and the identification function (Section 6.4) as the active-set prediction strategies.

In Chapter 7, we first briefly present the formulations of perturbed QP problems (Section 7.1) and show their properties which are similar to the linear case (Section 7.2). We then extend the prediction results to QP and derive theorems on predicting the optimal tripartition of a QP problem (Section 7.3). In Section 7.4, we present the structure of the perturbed algorithm for QP and the promising preliminary numerical experience.

In Chapter 8, we summarise the main contributions of this thesis and point out potential future research directions.

# 2

# Theoretical Aspects of Linear and Quadratic Programming

In this chapter, we present the theoretical background required by the results in this thesis. In Section 2.1, we briefly introduce the standard form of the primal-dual pair of LP problems, the structure of the solution set, the notion of degenerate problems and problems with multiple solutions. Section 2.2 focuses on some basic concepts of the primal-dual path-following interior point methods for LP, including the central path and its neighbourhoods, general algorithmic framework and implementation techniques. Then we briefly introduce the primal-dual pair of convex QP problems and present the difference between the structure of the solution set of QP and that of LP in Section 2.3. Finally in Section 2.4, we derive a global error bound for LP and QP. The error bounds obtained in this section measure the distance to the solution set of LP or QP problems, which plays an important role in our theoretical development, in Chapters 5 and 7.

## 2.1 Linear Programming

We consider the following pair of primal-dual LP problems,

$$\begin{array}{lll} \text{Primal} & & \text{Dual} \\ \min_x & c^T x & \max_{(y,s)} \quad b^T y \\ \text{s.t.} & Ax = b, & \text{s.t.} \quad A^T y + s = c, \\ & x \geq 0, & \qquad\quad s \geq 0, \end{array} \tag{PD}$$

where $A \in \mathbb{R}^{m\times n}$ has full row rank, $x, s, c \in \mathbb{R}^n$ and $y, b \in \mathbb{R}^m$ with $m \leq n$. (Note that the primal problem in (PD) in problem (1.1).) The primal and dual problems are closely related to each other in many ways, such as in the key duality result Theorem 2.1.

If a primal-dual pair $(x, y, s)$ satisfies $Ax = b$, $A^\top y + s = c$ and $(x, s) \geq 0$, it is a *primal-dual feasible point* of (PD). The set of all feasible points forms the *feasible set*, namely,

$$\mathcal{F} = \{(x, y, s) \,|\, Ax = b,\ A^T y + s = c,\ x \geq 0,\ s \geq 0\}. \tag{2.1}$$

A primal-dual feasible point is said to be *strictly feasible* if all components of $x$ and $s$

are positive. So we can define the strictly feasible set for (PD) as follows

$$\mathcal{F}^0 = \{(x, y, s) \,|\, Ax = b,\ A^T y + s = c,\ x > 0,\ s > 0\}. \tag{2.2}$$

Let $(x, y, s) \in \mathcal{F}$. Then we have

$$c^T x - b^T y = c^T x - (Ax)^T y = x^T (c - A^T y) = x^T s \geq 0.$$

Thus the quantity $x^T s$ measures the difference between the primal objective function and that of the dual at a feasible point. We refer to it as the *duality gap*. The gap is zero at an optimal solution of (PD) according to the next theorem.

**Theorem 2.1 (Duality for LP [135]).** *The primal problem in (PD) has a solution if and only if the dual problem has a solution and the optimal objective values are the same for the two problems.*

A different formulation of Theorem 2.1 follows next.

**Theorem 2.2.** *A primal-dual pair $(x^*, y^*, s^*)$ is a primal-dual optimal solution of (PD) if and only if it satisfies the following conditions,*

$$Ax = b, \tag{2.3a}$$
$$A^\top y + s = c, \tag{2.3b}$$
$$XSe = 0 \tag{2.3c}$$
$$(x, s) \geq 0, \tag{2.3d}$$

*where $x = (x_1, \ldots, x_n)$, $s = (s_1, \ldots, s_n)$, $X = diag(x)$, $S = diag(s)$ and $e$ is a vector of ones.*

These optimality conditions are a special case of the well known KKT conditions, which hold for general constrained optimisation [135]. It can be derived for (PD) according for example to [105, Theorem 12.1].

For the reminder of this thesis, we denote by $\Omega^P$ and $\Omega^D$ the primal and dual solution sets of the LP problems in (PD), respectively,

$$\Omega^P = \{x^* \,|\, x^* \text{ solves the primal problem in (PD) }\}, \tag{2.4a}$$
$$\Omega^D = \{(y^*, s^*) \,|\, (y^*, s^*) \text{ solves the dual problem in (PD) }\}. \tag{2.4b}$$

The *primal-dual solution set* $\Omega$ is the Cartesian product of $\Omega^P$ and $\Omega^D$ [135, Chapter

2], namely,

$$\Omega = \Omega^P \times \Omega^D = \{(x^*, y^*, s^*) \mid (x^*, y^*, s^*) \text{ satisfies (2.3)}\}.$$

Relation (2.3c) implies that at least one of $x_i^*$ and $s_i^*$ should be zero for all $i \in \{1, \ldots, n\}$. If $x_i^* + s_i^* > 0$, $\forall i \in \{1, \ldots, n\}$, we say that $(x^*, y^*, s^*)$ is a *strictly complementary solution* of (PD). The strictly complementary solution plays an important role in the construction of interior point methods. The following theorem shows that any primal-dual feasible LP problems have a strictly complementary solution.

**Theorem 2.3 (Goldman-Tucker Theorem [50]).** *If (PD) has a solution, then there exists at least one primal solution $x^* \in \Omega^P$ and one dual solution $(y^*, s^*) \in \Omega^D$ such that*

$$x^* + s^* > 0. \tag{2.5}$$

See [50] or [135, Chapter 2] for a proof using Farkas's Lemma .

Let $(x^*, y^*, s^*)$ be a (PD) solution. We employ the following notations

$$\mathcal{A}(x^*) := \{i \in \{1, \ldots, n\} \,|\, x_i^* = 0\}, \tag{2.6a}$$
$$\mathcal{A}^+(s^*) := \{\, i \in \{1, \ldots, n\} \,|\, s_i^* > 0\,\}, \tag{2.6b}$$
$$\mathcal{I}(s^*) := \{i \in \{1, \ldots, n\} \,|\, s_i^* = 0\}. \tag{2.6c}$$

$\mathcal{A}(x^*)$ and $\mathcal{A}^+(s^*)$ are referred to as the optimal (primal) active and strongly (primal) active sets at $(x^*, y^*, s^*)$; $\mathcal{I}(s^*)$ as the dual active set.

If $(x^*, y^*, s^*)$ is a strictly complementary solution of (PD), (2.5) implies that

$$\begin{aligned} \mathcal{A} &\equiv \mathcal{A}(x^*) = \mathcal{A}^+(s^*), \\ \mathcal{I} &\equiv \mathcal{I}(s^*) = \{1, \ldots, n\} \setminus \mathcal{A}^+(s^*) = \{1, \ldots, n\} \setminus \mathcal{A}(x^*). \end{aligned} \tag{2.7}$$

Thus

$$\mathcal{A}(x^*) \cap \mathcal{I}(s^*) = \emptyset \quad \text{and} \quad \mathcal{A}(x^*) \cup \mathcal{I}(s^*) = \{1, 2, \ldots, n\},$$

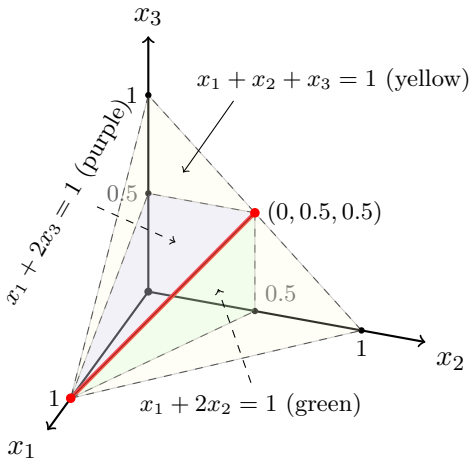
namely $\mathcal{A}(x^*)$ and $\mathcal{I}(s^*)$ form a *strict complementary partition* of the index set $\{1, \ldots, n\}$ for (PD). The strict complementary partition is the same for all strictly complementary solutions [138, Theorem 1.16] and so we can simplify our notations to $(\mathcal{A}, \mathcal{I})$ for the active-inactive strictly complementary partition.

**Degeneracy and multiplicity in the solution set.** Next, we describe the concept of LP problem degeneracy, as the local convergence properties of IPMs [63, 123] require problem nondegeneracy assumptions and more importantly, so do some of our theorems on active-set prediction. We use the same meaning of the terms 'nondegenerate' and

'degenerate' as in [135]. The definitions are given below; also see Figure 2.1 for an illustration.

**Definition 2.4 (Degeneracy for LP [135]).** *In the context of* LP*,*

- ***Primal degenerate*** *refers to the primal problem in* (PD) *having a solution* $x^*$ *containing less than* $m$ *positive components; the solution* $x^*$ *is called a* ***primal degenerate solution****.*
- ***Dual degenerate*** *refers to the dual problem in* (PD) *having a solution* $(y^*, s^*)$ *such that* $s^*$ *contains less than* $n-m$ *positive components; the solution* $(y^*, s^*)$ *is called a* ***dual degenerate solution****.*
- (PD) *is called* ***primal nondegenerate*** *if it is not primal degenerate; and* ***dual nondegenerate*** *if it is not dual degenerate.*




$$\begin{aligned} \min_{x\in\mathbb{R}^{3.5}} \quad & x_1 + 2x_3 \\ \text{subject to} \quad & \\ & x_1 + x_2 + x_3 = 1, \\ & x_1 + 2x_2 = 1, \\ & x_1 \geq 0, \quad x_2 \geq 0. \end{aligned}$$

It is clear that all points on the red line segment are optimal solutions of the given LP problem, among which $(1, 0, 0)$ is primal degenerate.

Figure 2.1: An illustration of the primal degeneracy for LP.

LP *multiplicity*, namely, that a given LP problem has multiple solutions, is similar and related to degeneracy. We summarise some of these relations in the following theorem.

**Theorem 2.5 (Multiplicity and degeneracy [119, Theorem 4.5]).** *For a pair of primal and dual linear programming problems in* (PD)*, the following implications hold:*

a. *If the primal problem has multiple solutions, then the dual is degenerate.*

b. *The primal problem has a unique and nondegenerate solution if and only if the dual has a unique and nondegenerate solution.*

c. *If the primal problem has multiple and nondegenerate solutions, then the dual has a unique and degenerate solution.*

*d. If the primal problem has a unique and degenerate solution, then the dual has multiple solutions.*

## 2.2 Primal-dual path-following IPM for LP

### 2.2.1 The primal-dual central path

To solve the barrier subproblem (1.3) of the LP problem (1.1) — or equivalently, of the primal problem in (PD) — we express its Lagrangian as follows,

$$L(x, y) = c^T x - \mu \sum_{j=1}^{n} \log x_j - y^T (Ax - b),$$

where $y \in \mathbb{R}^m$ is the Lagrangian multiplier and the first order optimality conditions for (1.3) are

$$\begin{aligned} \nabla_x L(x, y) &= c - A^T y - \mu X^{-1} e = 0, \\ \nabla_y L(x, y) &= Ax - b = 0, \end{aligned}$$

where $X = \operatorname{diag}(x)$ and $e$ is a vector of ones. Setting $s = \mu X^{-1} e$, we have

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ XSe &= \mu e, \\ (x, s) &\geq 0, \end{aligned} \tag{2.8}$$

where $S = \operatorname{diag}(s)$, and $\mu$ is the strictly positive barrier parameter. The system (2.8) can also be derived from the logarithmic barrier subproblem of the dual problem (1.2).

The only difference between (2.8) and (2.3) is the perturbation of the complementarity equations (2.3c), namely, instead of zero, the product $x_i s_i$ is set to $\mu > 0$ in (2.8). (2.8) has a unique solution under mild assumptions, including

**Assumption:** $A$ has full row rank $m$.[4] (2.9)

**Theorem 2.6 (Existence of the (PD) central path [135]).** *Let (2.9) hold. Then the system (2.8) has a unique solution for each $\mu > 0$, provided $\mathcal{F}^0$ in (2.2) is nonempty.*

See [135, Theorem 2.8] or [90] for a proof.

[4] This assumption is usually not difficult to ensure numerically; see Section 6.1.

When $\mu$ is fixed, the unique solution of (2.8), $(x(\mu), y(\mu), s(\mu))$ with $(x(\mu), s(\mu)) > 0$, is a primal-dual strictly feasible point of (PD). As $\mu$ decreases continuously, the corresponding solution $(x(\mu), y(\mu), s(\mu))$ of (2.8) will form a path $\mathcal{C}_{PD}$ of primal-dual strictly feasible points, which is the *central path* of (PD). And when $\mu$ approaches 0, $(x(\mu), y(\mu), s(\mu))$ will approach a strictly complementary solution of the (PD) problems.

**Theorem 2.7 (Convergence of the central path [138, Theorem 2.14]).** *Let (2.9) hold and $\mathcal{F}^0 \neq \emptyset$. As $\mu \to 0$, the unique solution $(x(\mu), y(\mu), s(\mu))$ of (2.8) converges to a strictly complementary solution of (PD).*

Unlike the primal central path $\mathcal{C}_P$ introduced in Section 1.3, the path $\mathcal{C}_{PD}$ contains both primal and dual points, so it is also called the primal-dual central path. For the rest of this thesis, the term 'central path' always refers to the 'primal-dual central path'.

**Neighbourhoods of the central path.** As mentioned before, primal-dual path-following IPMs generate iterates that follow the central path. It is not recommended to generate iterates exactly on the central path because finding a point that solves (2.8) can be as difficult as solving the optimisation problem itself [24]. Therefore we only require the iterates to lie in some neighbourhood of the central path. To achieve this, different lower and/or upper bounds are applied to the pairwise complementarity products, namely $x_i s_i$ for $i = 1, \ldots, n$. In theoretical developments, one of the most commonly used neighbourhoods is the one sided $l_\infty$-norm (wide) neighbourhood, defined by

$$\mathcal{N}_{-\infty}(\gamma) = \{ (x, y, s) \in \mathcal{F}^0 \mid x_i s_i \geq \gamma\mu,\ i = 1, \ldots, n \} \tag{2.10}$$

with given $\gamma \in (0, 1)$, where $\mathcal{F}^0$ defined in (2.2) is the strictly feasible set of (PD) and $\mu$ is equivalent to the average complementarity products (i.e. $\mu = \frac{x^T s}{n}$). The use of the $\mathcal{N}_{-\infty}(\gamma)$ neighbourhood is to keep the pairwise complementarity products bounded from below (with respect to their mean value) and so prevent the components of $x$ and $s$ from being too close to zero when their mean value is not. $\mathcal{N}_{-\infty}(\gamma)$ is a wide neighbourhood and can enclose all points in $\mathcal{F}^0$ by pushing $\gamma$ to 0.

In addition to the above neighbourhood, Gondzio [52] also introduced the following symmetric neighbourhood,

$$\mathcal{N}_s(\gamma) = \{ (x, y, s) \in \mathcal{F}^0 \mid \gamma\mu \leq x_i s_i \leq \frac{\mu}{\gamma},\ i = 1, \ldots, n \}, \tag{2.11}$$

where $\gamma \in (0, 1)$ and $\mu$ is also the average complementarity products. Within the $\mathcal{N}_s(\gamma)$ neighbourhood the pairwise complementarity is bounded below and also above.

Note that for the ease of theoretical development, the neighbourhoods mentioned above require the iterates to be primal-dual feasible. Thus, methods developed with these neighbourhoods fall into the category of *feasible primal-dual path-following* IPMs.

In practice, the *infeasible primal-dual path-following* IPMs are more commonly used, which allow iterates to violate the equality constraints ($(x, s)$ still need to be positive). In the following section, we describe the general framework for this type of IPMs.

### 2.2.2 A general primal-dual path-following IPM framework

Let $(x^k, y^k, s^k)$, $k \geq 0$, be the current iterate of the IPM applying to the (PD) problems and having $(x^k, s^k) > 0$. We apply Newton's method to (2.3) to determine the search direction $(\Delta x^k, \Delta y^k, \Delta s^k)$, namely we solve the following system of equations,

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^\top & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = - \begin{bmatrix} Ax^k - b \\ A^\top y^k + s^k - c \\ X^k S^k e - \mu e \end{bmatrix}, \tag{2.12}$$

where $X^k = \text{diag}(x^k)$ and $S^k = \text{diag}(s^k)$. A full step along this direction may violate the constraints $(x, s) \geq 0$, so we need to do a line search to decide the stepsize. After one step, the IPMs shrink $\mu$ by a factor $\sigma \in (0, 1)$, which is called the *centering parameter*. Now we show the general structure of a primal-dual path-following IPM in Algorithm 2.1.

**Algorithm 2.1** A Primal-Dual Path-Following Interior Point Framework [135]

**Given** $(x^0, y^0, s^0)$ with $(x^0, s^0) > 0$;
**for** $k = 0, 1, 2, \ldots$ **do**
  **Choose** $\sigma^k \in (0, 1)$ and obtain a Newton step $(\Delta x^k, \Delta y^k, \Delta s^k)$ by solving the system (2.12) with $\mu = \sigma^k \mu^k$ and

$$\mu^k = \frac{(x^k)^\top s^k}{n}; \tag{2.13}$$

  **Choose** a step length $\alpha^k \in (0, 1]$, which sufficiently reduces the duality gap and the residuals of equality constraints and also keeps $(x^{k+1}, s^{k+1}) > 0$;
  **Set** $(x^{k+1}, y^{k+1}, s^{k+1}) = (x^k, y^k, s^k) + \alpha^k (\Delta x^k, \Delta y^k, \Delta s^k)$.
**end for**

**Starting point.** A good starting point is essential for primal-dual IPMs to make good progress. We adopt the popular method of finding a starting point proposed by Mehrotra [93]. Let (2.9) hold. First obtain $(\tilde{x}, \tilde{y}, \tilde{s})$ by

$$\tilde{x} = A^T (AA^T)^{-1} b, \quad \tilde{y} = (AA^T)^{-1} Ac, \quad \tilde{s} = c - A^T \tilde{y},$$

and then we compute the starting point as follows,

$$(x^0, y^0, s^0) = (\tilde{x} + \tilde{\delta}_x e,\ \tilde{y},\ \tilde{s} + \tilde{\delta}_s e),$$

where

$$\tilde{\delta}_x = \delta_x + 0.5 \times \frac{(\tilde{x} + \delta_x e)^T (\tilde{s} + \delta_s e)}{\sum_{i=1}^{i=n} (\tilde{s}_i + \delta_s)} \quad \text{and} \quad \tilde{\delta}_s = \delta_s + 0.5 \times \frac{(\tilde{x} + \delta_x e)^T (\tilde{s} + \delta_s e)}{\sum_{i=1}^{i=n} (\tilde{x}_i + \delta_x)},$$

and where $\delta_x = \max\left(-1.5 \times \min_{i=1,\ldots,n} (\tilde{x}_i), 0\right)$ and $\delta_s = \max\left(-1.5 \times \min_{i=1,\ldots,n} (\tilde{s}_i), 0\right)$.

**Solving the Newton system.** Eliminating $\Delta s^k$ in (2.12) with $\mu = \sigma^k \mu^k$, we have

$$\begin{bmatrix} A & 0 \\ -D^{-2} & A^T \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \end{bmatrix} = - \begin{bmatrix} R_p^k \\ R_d^k - (X^k)^{-1} R_c^k \end{bmatrix}, \tag{2.14a}$$

$$\Delta s^k = - \left(X^k\right)^{-1} \left(R_c^k + S^k \Delta x^k\right). \tag{2.14b}$$

where $D = -\left(S^k\right)^{-\frac{1}{2}} \left(X^k\right)^{\frac{1}{2}}$, $R_p^k = Ax^k - b$, $R_d^k = A^T y^k + s^k - c$, $R_c^k = X^k S^k e - \sigma^k \mu^k e$ and where

$$\sigma^k = \min(0.1, 100\mu^k) \in [0, 1] \tag{2.15}$$

and $\mu^k$ is defined in (2.13). The system (2.14) is called the *augmented system* [135].

We can further eliminated $\Delta x^k$ from (2.14a) and then (2.14) is equivalent to

$$A(D^k)^2 A^T \Delta y^k = -R_p^k - A(S^k)^{-1}(X^k R_d^k - R_\mu^k), \tag{2.16a}$$

$$\Delta s^k = -R_d^k - A^T \Delta y^k, \tag{2.16b}$$

$$\Delta x^k = -(S^k)^{-1}(R_\mu^k + X^k \Delta s^k). \tag{2.16c}$$

This system is known as the *normal equations*. The normal equations form has been widely used in both commercial and academic IPM codes that require to handle large-scale problems efficiently, because the coefficient matrix of $\Delta y^k$ in (2.16a) can be efficiently processed by existing advanced numerical linear algebra techniques, such as efficient Cholesky factorisations; in our experimental codes, however, we use the augmented system because it is more stable in the presence of ill-conditioning [135].

**Stepsizes in the primal and dual space.** For efficiency, instead of imposing the neighbourhood constraints (2.10) or (2.11), one can choose the stepsize as a fixed fraction of the step to the nearest constraint boundary in the primal and dual space, respectively. Namely, we compute possibly distinct stepsizes $\alpha_p^k$ for the primal iterates and $\alpha_d^k$ for the dual ones as follows

$$\alpha_p^k = \min\left(\bar{\alpha} \cdot \min_{i:\Delta x_i^k < 0} \left(\frac{-x_i^k}{\Delta x_i^k}\right), 1\right) \quad \text{and} \quad \alpha_d^k = \min\left(\bar{\alpha} \cdot \min_{i:\Delta s_i^k < 0} \left(\frac{-s_i^k}{\Delta s_i^k}\right), 1\right), \tag{2.17}$$

where $\bar{\alpha} \in (0,1)$; usually $\bar{\alpha} = 0.9995$. Although such a variant of algorithm may not converge,[5] this strategy for selecting stepsizes has been used in many practical implementations [105, Section 14.2] and proven effective in practice [36].

**Termination.** We measure the relative residual by

$$\text{relRes}^k = \frac{|| \left(Ax^k - b, A^T y^k + s^k - c, X^k S^k e\right) ||_\infty}{1 + \max\left(||b||_\infty, ||c||_\infty\right)}. \tag{2.18}$$

We can terminate the algorithm when this relative residual is less than the required accuracy. Recalling (2.3), a small value of $\text{relRes}^k$ indicates we are (relatively) nearly optimal for (PD).

### 2.2.3 Bounds on the sequence of iterates

In this section, we discuss two lemmas which are essential to our theoretical developments later on in this thesis. These lemmas are originally two parts of Lemma 5.13 in [135]. The proofs here follow similarly to the proof of [135, Lemma 5.13]. We separate the original result into two parts in order to suit our needs. Note that part of the original proof requires weaker conditions than the other, which is reflected in the assumptions of the following lemmas.

**Lemma 2.8 ([135, Lemma 5.13]).** *For any $(x, y, s) \in \mathcal{F}^0$, where $\mathcal{F}^0$ is defined in (2.2), we have*

$$0 < x_i \leq \frac{\mu}{C_1} \quad (i \in \mathcal{A}) \quad \text{and} \quad 0 < s_i \leq \frac{\mu}{C_1} \quad (i \in \mathcal{I}), \tag{2.19}$$

*where*

$$\mu = \frac{x^T s}{n} \tag{2.20}$$

*and*

$$C_1 = \frac{\epsilon(A, b, c)}{n} \tag{2.21}$$

*with*

$$\epsilon(A, b, c) = \min\left( \min_{i \in \mathcal{I}} \sup_{x^* \in \Omega^P} \{ x_i^* \}, \min_{i \in \mathcal{A}} \sup_{(y^*, s^*) \in \Omega^D} \{ s_i^* \} \right) > 0, \tag{2.22}$$

*and $\Omega^P$ and $\Omega^D$, defined in (2.4), are the primal and dual solution sets of (PD), respectively, and where $(\mathcal{A}, \mathcal{I})$ is the strictly complementary active and inactive partition (2.7) of the solution set of (PD).*

**Proof.** Assume $(x^*, y^*, s^*)$ is any (PD) solution. Since $(x, y, s)$ is feasible, we have $Ax = Ax^* = b$ and $A^T y - s = A^T y^* - s^* = c$, and so $A(x - x^*) = 0$ and $s - s^* = A^T(y - y^*)$. This gives us

$$(x - x^*)^T (s - s^*) = \left(A(x - x^*)\right)^T (y - y^*) = 0. \tag{2.23}$$

[5] In order to have the polynomial complexity, the stepsizes should be chosen such that all iterates are within a certain neighbourhood of the central path.

Since $(\mathcal{A},\mathcal{I})$ is the strictly complementary partition, $x_i^* = 0$ for all $i \in \mathcal{A}$ and $s_i^* = 0$ for $i \in \mathcal{I}$. From this, (2.20) and (2.23) we have

$$n\mu = x^T s = x^T s^* + s^T x^* = \sum_{i\in\mathcal{A}} x_i s_i^* + \sum_{i\in\mathcal{I}} s_i x_i^*.$$

From the nonnegativity of each term in the above summations, we obtain $0 < x_i < \frac{n\mu}{s_i^*}$ for all $i \in \mathcal{A}$, and $0 < s_i < \frac{n\mu}{x_i^*}$ for all $i \in \mathcal{I}$, which implies

$$0 < x_i < \frac{n\mu}{\sup_{(y^*,s^*)\in\Omega^D} s_i^*} \quad \text{and} \quad 0 < s_i < \frac{n\mu}{\sup_{x^*\in\Omega^P} x_i^*},$$

and since $(x^*, y^*, s^*)$ is any (PD) solution, we have

$$0 < \max_{i\in\mathcal{A}} x_i < \frac{n\mu}{\min_{i\in\mathcal{A}} \sup_{(y^*,s^*)\in\Omega^D} s_i^*} \quad \text{and} \quad 0 < \max_{i\in\mathcal{I}} s_i < \frac{n\mu}{\min_{i\in\mathcal{I}} \sup_{x^*\in\Omega^P} x_i^*}.$$

Combining these two sets of inequalities and recalling (2.22), we have

$$0 < \max\left(\max_{i\in\mathcal{A}} x_i, \max_{i\in\mathcal{I}} s_i\right) < \frac{n\mu}{\epsilon(A,b,c)},$$

which implies (2.19). Note that when the feasible set of (PD) is bounded and nonempty, where $\epsilon(A,b,c) > 0$. □

**Lemma 2.9 ([135, Lemma 5.13]).** *For any $(x,y,s) \in \mathcal{N}_{-\infty}(\gamma)$, where $\mathcal{N}_{-\infty}(\gamma)$ is defined in (2.10), we have*

$$s_i \geq C_1\gamma \quad (i \in \mathcal{A}) \quad \text{and} \quad x_i \geq C_1\gamma \quad (i \in \mathcal{I}), \tag{2.24}$$

*where $\mu$ is defined in (2.20), $C_1$ in (5.5) and $(\mathcal{A},\mathcal{I})$ is the strictly complementary active and inactive partition (2.7) of the solution set of (PD).*

**Proof.** From $(x,y,s) \in \mathcal{N}_{-\infty}(\gamma)$, (2.10) and (2.19) we have

$$x_i \geq \frac{\mu\gamma}{s_i} \geq \frac{\mu\gamma}{\frac{\mu}{C_1}} = \gamma C_1, \text{ for all } i \in \mathcal{I}.$$

Similarly for $s_i$, $i \in \mathcal{A}$. □

## 2.3 Quadratic programming

Consider the following pair of primal and dual **convex** QP problems,

$$\begin{array}{ll|ll} \text{(Primal)} & & \text{(Dual)} & \\ \min\limits_{x} & \frac{1}{2}x^THx + c^Tx & \max\limits_{(x,y,s)} & b^Ty - \frac{1}{2}x^THx \\ \text{s.t.} & Ax = b, & \text{s.t.} & A^Ty + s - Hx = c, \\ & x \geq 0, & & y \text{ free},\ s \geq 0, \end{array} \tag{QPD}$$

where $H \in \mathbb{R}^{n\times n}$ is symmetric positive semidefinite, $A \in \mathbb{R}^{m\times n}$ with $m \leq n$, $y$, $b \in \mathbb{R}^m$ and $x$, $s$, $c \in \mathbb{R}^n$. When $H$ is an empty matrix, these problems reduce to the (PD) linear programming problems (see page 16).

The feasible set of (QPD) is denoted as

$$\mathcal{QF} = \{(x, y, s) \,|\, Ax = b,\ A^Ty + s - Hx = c,\ x \geq 0,\ s \geq 0\}, \tag{2.25}$$

and the strictly feasible set as

$$\mathcal{QF}^0 = \{(x, y, s) \,|\, Ax = b,\ A^Ty + s - Hx = c,\ x > 0,\ s > 0\}. \tag{2.26}$$

Similarly to LP, we can derive the following KKT conditions for (QPD),

$$Ax = b, \tag{2.27a}$$
$$A^Ty + s - Hx = c, \tag{2.27b}$$
$$XSe = 0, \tag{2.27c}$$
$$(x, s) \geq 0, \tag{2.27d}$$

where $X = \text{diag}(x)$, $S = \text{diag}(s)$ and $e$ is a vector of ones. The third term is called the complementary condition [54].

**Structure of the solution set of (QPD).** We denote the solution set of the primal and dual problems in (QPD) as $Q\Omega^P$ and $Q\Omega^D$, respectively, and the primal-dual solution set $Q\Omega$ of (QPD) is also a Cartesian production of $Q\Omega^P$ and $Q\Omega^D$. Contrary to linear programming, the Goldman–Tucker Theorem does not hold for (QPD), namely it is not guaranteed to have a strictly complementary solution for (QPD).

Let $(x^*, y^*, s^*)$ be a solution of (QPD) and define

$$\begin{aligned} \mathcal{A}(x^*) &= \{i \in \{1, \ldots, n\} \,|\, x_i^* = 0\}, \quad \Theta(x^*) = \{i \in \{1, \ldots, n\} \,|\, x_i^* > 0\}, \\ \mathcal{I}(s^*) &= \{i \in \{1, \ldots, n\} \,|\, s_i^* = 0\}, \quad \mathcal{A}^+(s^*) = \{i \in \{1, \ldots, n\} \,|\, s_i^* > 0\}. \end{aligned} \tag{2.28}$$

$\mathcal{A}(x^*)$ is the *primal active set* of (QPD), $\Theta(x^*)$ the *primal inactive set*, $\mathcal{I}(s^*)$ the *dual active set* and $\mathcal{A}^+(s^*)$ the *dual inactive set*. From the complementary condition (2.27c),

it is easy to verify that

$$\mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*), \quad \Theta(x^*) \subseteq \mathcal{I}(s^*) \quad \text{and} \quad \Theta(x^*) \cap \mathcal{A}^+(s^*) = \emptyset. \tag{2.29}$$

Note that $\mathcal{A}(x^*) \cap \mathcal{I}(s^*)$ may not be empty.

We also denote

$$\mathcal{T}(x^*, s^*) = \{1, \ldots, n\} \setminus \left(\mathcal{A}^+(s^*) \cup \Theta(x^*)\right), \tag{2.30}$$

which represents the complement of the optimal primal and dual inactive sets. This and (2.29) give us that $\mathcal{A}^+(s^*) \cap \Theta(x^*) = \mathcal{A}^+(s^*) \cap \mathcal{T}(x^*, s^*) = \Theta(x^*) \cap \mathcal{T}(x^*, s^*) = \emptyset$, and the union of them is the full index set, namely, $\mathcal{A}^+(s^*)$, $\Theta(x^*)$ and $\mathcal{T}(x^*, s^*)$ form an *optimal tripartition* of $\{1, \ldots, n\}$ for (QPD). From the definition of $\mathcal{T}(x^*, s^*)$, we have $x_i^* = s_i^* = 0$ for any $i \in \mathcal{T}(x^*, s^*)$ and thus it is also straightforward to verify

$$\mathcal{A}(x^*) = \mathcal{A}^+(s^*) \cup \mathcal{T}(x^*, s^*) \quad \text{and} \quad \mathcal{I}(s^*) = \Theta(x^*) \cup \mathcal{T}(x^*, s^*).$$

The primal-dual pair in (QPD) always has a *maximal complementary solution*, at which the number of positive components of $x^* + s^*$ is maximised [64]. Even at a maximal complementary solution, $\mathcal{T}(x^*, s^*)$ may not be empty because of the absence of the Goldman–Tucker Theorem for (QPD). Note that $(\mathcal{A}^+(s^*), \Theta(x^*), \mathcal{T}(x^*, s^*))$ forms a tripartition at any solution of (QPD) but it may be different at different solutions; the tripartitions are only guaranteed to be invariant at maximal complementary solutions [138, Theorem 1.18].

**Interior point methods for quadratic programming.** The IPMs for QP is a natural extension from that for LP. Besides solving a slightly different Newton system, the algorithm follows the same structure as IPMs for LP. Because of the presence of the quadratic terms in (QPD), we have one extra matrix $H$, the coefficient matrix of the quadratic terms, in the dual feasibility equation in (2.27), which leads to solving the following Newton system,

$$\begin{bmatrix} A & 0 & 0 \\ -H & A^\top & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = - \begin{bmatrix} Ax^k - b \\ A^\top y^k + s^k - Hx - c \\ X^k S^k e - \mu e \end{bmatrix}. \tag{2.31}$$

The augmented system can also be derived by eliminating $\Delta s^k$; normal equations form can be obtained by further eliminating $\Delta x^k$. The best known IPM algorithm for QP finds the $\epsilon$-accurate solution of a convex QP problem in $\mathcal{O}\left(\sqrt{n}\ln(1/\epsilon)\right)$ iterations [54].

## 2.4 Error bounds for linear and quadratic programming

Error bounds for an optimisation problem bound the distance from a given point to the solution set of the problem in terms of a residual function. In the implementation of iterative optimisation methods, error bounds can be applied to estimate the distance from the current iterate to the (unknown) optimal face, using local information at the current iterate [108]. Thus error bounds can sometimes be effective for obtaining termination criteria. Furthermore, error bounds can be used to identify the active set by means of an identification function [35]; see for example [29], where the authors employ error bounds to construct an identification function so as to predict the active sets of specific QP problems arising from the area of machine learning (support vetoer machines).

In this section, we first formulate the (QPD) problem as a monotone Linear Complementarity Problem (LCP) and then apply a global error bound for the monotone LCP to the reformulated QP problem in order to derive an error bound for (QPD); then, by setting $H \equiv 0_{n\times n}$, where $0_{n\times n}$ stands for $n$-dimensional zero matrix, we obtain similar results for LP.

The error bound we employ in this section was proposed by Mangasarian and Ren in [85]. In their paper, they compared several global error bounds and concluded that the error bound we describe here, which measures the average of two different residuals, can be considered as the best. We also looked at other possibilities, such as the componentwise error bounds by Wang and Yuan [128], which only works under certain conditions (such as $H$-matrix with positive diagonal components). Unfortunately, our LCP formulation of (QPD) does not satisfy these conditions.

### 2.4.1 An error bound for QP

By setting $s = c - A^\top y + Hx$ and $y = y^+ - y^-$, where $y^+ = \max(y, 0)$ and $y^- = -\min(y, 0)$, the first order optimality conditions (2.27) for (QPD) can be reformulated as

$$\begin{aligned} Ax - b \geq 0,\ -Ax + b \geq 0,\\ c - A^\top y^+ + A^\top y^- + Hx \geq 0,\\ x^T(c - A^\top y^+ + A^\top y^- + Hx) = 0,\\ x \geq 0,\ y^+ \geq 0,\ y^- \geq 0. \end{aligned} \tag{2.32}$$

Let

$$M = \begin{bmatrix} H & -A^T & A^T \\ A & 0 & 0 \\ -A & 0 & 0 \end{bmatrix}, \quad q = \begin{bmatrix} c \\ -b \\ b \end{bmatrix} \quad \text{and} \quad z = \begin{bmatrix} x \\ y^+ \\ y^- \end{bmatrix}. \tag{2.33}$$

where $H$, $A$, $b$ and $c$ are (QPD) problem data, $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$. Then finding

a solution of (2.32) is equivalent to solving the following problem,

$$Mz + q \geq 0, \quad z \geq 0, \quad z^T(Mz + q) = 0, \tag{2.34}$$

where $M$, $q$ and $z$ are defined in (2.33), and $z$ is considered to be the vector of variables. Given the above relations, the equivalence of the QP solution set and that of its corresponding LCP formulation is straightforward.

**Lemma 2.10.** *(QPD) is equivalent to the* LCP *in (2.34) with $M$ and $q$ defined in (2.33), namely,*

1. *If $(x, y^+, y^-)$ is a solution of the* LCP *(2.34), then $(x, y, s)$ is a (QPD) solution, where $y = y^+ - y^-$ and $s = c - A^T y + Hx$.*
2. *If $(x, y, s)$ is a (QPD) solution, then $(x, y^+, y^-)$ is a solution of the* LCP *(2.34).*

We can see that (2.34) is in the standard form of a classic LCP, which was studied by Cottle and Dantzig in 1968 [26]. LCP problems can be divided into different classes depending on the type of the matrix $M$. An LCP is called the *monotone* LCP if and only if the matrix $M$ is positive semidefinite. For definitions and properties of the other classes of LCP, see [26]. Next we show that (QPD) can be viewed as a monotone LCP [26].

**Lemma 2.11.** *The matrix $M$, defined in (2.33), is positive semidefinite, and so (2.34) is a monotone* LCP.

**Proof.** $\forall v \neq 0, v = (v_1, v_2, v_3)$, where $v_1 \in \mathbb{R}^n, v_2 \in \mathbb{R}^m$ and $v_3 \in \mathbb{R}^m$. $v^T M v = v_1^T H v_1 + v_2^T A v_1 - v_3^T A v_1 - v_1^T A^T v_2 + v_1^T A^T v_3$. Since $v_2^T A v_1 = (v_2^T A v_1)^T = v_1^T A^T v_2$ and $v_3^T A v_1 = (v_3^T A v_1)^T = v_1^T A^T v_3$, we have $v^T M v = v_1^T H v_1 \geq 0$ as $H$ is positive semidefinite. Thus $M$ is positive semidefinite. □

In [85], the authors have proved a global error bound for the monotone LCP (2.34). Now we summarise their result in the following lemma.

**Lemma 2.12 (Mangasarian and Ren [85, Corollary 2.2]).** *Let $z$ be any point away from the solution set of a monotone* LCP*$(M,q)$ (2.34) and $z^*$ be the closest solution of (2.34) to $z$ under the Euclidean norm $\|\cdot\|$. Then $r(z) + w(z)$ is a global error bound for (2.34), namely,*

$$\|z - z^*\| \leq \tau(r(z) + w(z)),$$

*where $\tau$ is some problem-dependent constant, independent of $z$ and $z^*$, and*

$$r(z) = \|z - (z - Mz - q)_+\| \quad \text{and} \quad w(z) = \|\left(-Mz - q, -z, z^T(Mz + q)\right)_+\|. \tag{2.35}$$

Next we deduce the error bound for our monotone LCP formulation (2.34).

**Lemma 2.13.** *Given the monotone* LCP *(2.34) with $M$ and $q$ defined in (2.33), let $(x, y^+, y^-)$ be any point away from the solution set of this problem and $(x^*, (y^*)^+, (y^*)^-)$*

*be the closest solution of this* LCP *to* $(x, y^+, y^-)$ *under the Euclidean norm* $\|\cdot\|$. *Then we have*

$$\|(x, y^+, y^-) - (x^*, (y^*)^+, (y^*)^-)\| \leq \tau(r(x, y^+, y^-) + w(x, y^+, y^-)),$$

*where* $\tau$ *is some problem-dependent constant, independent of* $(x, y^+, y^-)$ *and* $(x^*, (y^*)^+, (y^*)^-)$,

$$r(x, y^+, y^-) = \left\| \left( \min\left\{x, c - A^T y + Hx\right\},\ \min\left\{y^+, Ax - b\right\},\ \min\left\{y^-, b - Ax\right\} \right) \right\|,$$

*and*

$$w(x, y^+, y^-) = \|(-(c - A^T y + Hx),\ b - Ax,\ Ax - b,\ -x,\ -y^+,\ -y^-,\ c^T x - b^T y + x^T Hx)_+\|,$$

*and where* $\min\{x, s\} = (\min(x_i, s_i))_{i=1,\ldots,n}$ *and* $y = y^+ - y^-$.

**Proof.** Substituting (2.33) into (2.35) and noting that $u - (u - v)_+ = \min\{u, v\}$ for any $u$, $v$ vectors, we have

$$\begin{aligned} &r(x, y^+, y^-) \\ &= \left\|\left(x - (x - (c - A^T(y^+ - y^-) + Hx))_+,\ y^+ - (y^+ - (Ax - b))_+,\ y^- - (y^- - (b - Ax))_+\right)\right\|, \end{aligned}$$

and

$$\begin{aligned} &w(x, y^+, y^-) \\ &= \left\|\left(-(c - A^T(y^+ - y^-) + Hx),\ b - Ax,\ Ax - b,\ -x,\ -y^+,\ -y^-,\ c^T x - b^T(y^+ - y^-) + x^T Hx\right)_+\right\|. \end{aligned}$$

Recalling $y = y^+ - y^-$, the lemma follows directly from the above equations. □

Next we show that $r(z) + w(z)$ is a global error bound for the (QPD) problems.

**Theorem 2.14 (Error bound for (QPD)).** *Let* $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ *where* $s = c - A^T y + Hx$. *Then there exist a solution* $(x^*, y^*, s^*)$ *of* (QPD) *and problem-dependent constants* $\tau_p$, $\tau_y$ *and* $\tau_d$, *independent of* $(x, y, s)$ *and* $(x^*, y^*, s^*)$, *such that*

$$\begin{aligned} \|x - x^*\| &\leq \tau_p \left(r(x, y, s) + w(x, y, s)\right), \\ \|y - y^*\| &\leq \tau_y \left(r(x, y, s) + w(x, y, s)\right), \\ \|s - s^*\| &\leq \tau_d \left(r(x, y, s) + w(x, y, s)\right), \end{aligned}$$

*where*

$$r(x, y, s) = \left\|\left(\min\{x, s\},\ \min\left\{y^+, Ax - b\right\},\ \min\left\{y^-, -Ax + b\right\}\right)\right\|, \tag{2.36}$$

*and*

$$w(x,y,s) = \|(-s,\ b-Ax,\ Ax-b,\ -x,\ c^Tx - b^Ty + x^THx)_+\|, \tag{2.37}$$

*and where* $\min\{x,s\} = (\min(x_i,s_i))_{i=1,\ldots,n}$, $y^+ = \max\{y,0\}$ *and* $y^- = -\min\{y,0\}$.

**Proof.** Consider the monotone LCP (2.34) with $M$ and $q$ defined in (2.33) and $z = (x, y^+, y^-)$. Let $z^* = (x^*, (y^*)^+, (y^*)^-)$ be the closest solution to $z$ in the solution set of this LCP. From Lemma 2.10, $(x^*, y^*, s^*)$ with $y^* = (y^*)^+ - (y^*)^-$ and $s^* = c - A^Ty^* + Hx^*$ is a (QPD) solution. From $(y^+, y^-) \geq 0$, $s = c - A^Ty + Hx$ and Lemma 2.13, we have

$$\|(x, y^+, y^-) - (x^*, (y^*)^+, (y^*)^-)\| \leq \tau(r(x,y,s) + w(x,y,s)),$$

where $r(x,y,s)$ and $w(x,y,s)$ are defined in (2.36) and (2.37), respectively. This and norm properties give

$$\max\left(\|x-x^*\|, \|y^+ - (y^*)^+\|, \|y^- - (y^*)^-\|\right) \leq \tau(r(x,y,s) + w(x,y,s)),$$

and so letting $\tau_p = \tau$, we deduce $\|x - x^*\| \leq \tau_p(r(x,y,s) + w(x,y,s))$. Also

$$\|y - y^*\| \leq \|y^+ - (y^*)^+\| + \|y^- + (y^*)^-\| \leq \tau_y(r(x,y,s) + w(x,y,s)),$$

where $\tau_y = 2\tau$. Since $s^* = c - A^Ty^* + Hx^*$, we also have

$$\begin{aligned} \|s - s^*\| &\leq \|A^T\|\|y - y^*\| + \|H\|\|x - x^*\| \\ &\leq \|A^T\|(\|y^+ - (y^*)^+\| + \|y^- - (y^*)^-\|) + \|H\|\|x - x^*\| \\ &\leq \tau_d(r(x,y,s) + w(x,y,s)), \end{aligned}$$

where $\tau_d = \tau(2\|A^T\| + \|H\|)$. □

Note that in Lemma 2.12, the solution $z^*$ is the closest solution to $z$, but here in Theorem 2.14, we may lose the property that $(x^*, y^*, s^*)$ is the closest solution to the given point in the Euclidean norm.

### 2.4.2 An error bound for LP

Setting $H \equiv 0_{n\times n}$, we can obtain similar results for LP. For clarity, we state them next. Their proofs follow by setting $H \equiv 0_{n\times n}$ in Lemma 2.10 and Theoem 2.14.

**Lemma 2.15.** (PD) *is equivalent to the* LCP *in* (2.34) *with* $M$ *and* $q$ *defined in* (2.33) *and* $H \equiv 0_{n\times n}$, *namely,*

1. *If* $(x, y^+, y^-)$ *is a solution of the* LCP (2.34)*, then* $(x, y, s)$ *is a* (PD) *solution, where* $y = y^+ - y^-$ *and* $s = c - A^T y$.

2. *If* $(x, y, s)$ *is a* (PD) *solution, then* $(x, y^+, y^-)$ *is a solution of the* LCP (2.34).

**Theorem 2.16 (Error bound for LP).** *Let* $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ *where* $s = c - A^T y$. *Then there exist a* (PD) *solution* $(x^*, y^*, s^*)$ *and problem-dependent constants* $\tau_p$, $\tau_y$ *and* $\tau_d$, *independent of* $(x, y, s)$ *and* $(x^*, y^*, s^*)$, *such that*

$$\begin{aligned} \|x - x^*\| &\leq \tau_p \left(r(x, y, s) + w(x, y, s)\right), \\ \|y - y^*\| &\leq \tau_y \left(r(x, y, s) + w(x, y, s)\right), \\ \|s - s^*\| &\leq \tau_d \left(r(x, y, s) + w(x, y, s)\right), \end{aligned} \tag{2.38}$$

*where*

$$r(x, y, s) = \left\| \left( \min\{x, s\}, \ \min\left\{y^+, Ax - b\right\}, \ \min\left\{y^-, -Ax + b\right\} \right) \right\|, \tag{2.39}$$

*and*

$$w(x, y, s) = \|(-s, \ b - Ax, \ Ax - b, \ -x, \ c^T x - b^T y)_+\|, \tag{2.40}$$

*and where* $\min\{x, s\} = (\min(x_i, s_i))_{i=1,\ldots,n}$, $y^+ = \max\{y, 0\}$ *and* $y^- = -\min\{y, 0\}$.

# 3 Active-set Prediction Strategies for Interior Point Methods

In Section 3.1, we apply the error bound introduced in Theorem 2.16 to construct an identification function [35] for linear programming and prove that this identification function can identify/predict the optimal active set when the error bound is small enough (which means the iterates are close enough to the optimal face). In Sections 3.2 and 3.3, we present the technical details of indicators and simple cut-off procedures respectively. We end this chapter by numerically comparing the performance of the identification function, indicators (primal-dual and Tapia indicators) and cut-off techniques, when used in an (infeasible) primal-dual path-following IPM (Algorithm 2.1) on standard LP test problems.

## 3.1 An identification function for linear programming

In [35], the authors present an active-set identification/prediction technique for inequality constrained nonlinear programming problems, relying upon a so-called identification function, a function that tends to zero when approaching the solution set but at a 'slower' rate than the Euclidean distance from the solution set. However, their idea and presentation are for NLP and no explicit formulation of the identification function for LP is given. Though our approach is similar to that of [29], their proposed identification function is specifically developed for the particular QP formulation of a support vector machine problem, and it seems their results cannot be applied to our (PD) problems by simple modifications such as setting quadratic terms to zero.

In this section, we construct a function based on the error bound (Section 2.4.2) for (PD) problems. Since our function has similar properties to the identification function mentioned above, we keep using the term 'identification function'. We define the following function,

$$\rho(x, y, s) := (r(x, y, s) + w(x, y, s))^{\frac{1}{2}}, \tag{3.1}$$

where $r(x, y, s)$ and $w(x, y, s)$ are defined in (2.39) and (2.40), respectively.

**Properties of the identification function (3.1).** It is clear that $\rho(x, y, s)$ in (3.1) is continuous in $(x, y, s)$. Now we show that the $\rho(x, y, s)$ is zero at a (PD) solution.

**Proposition 3.1.** *Assume $(x^*, y^*, s^*)$ is a (PD) solution. Then*

$$\rho\left(x^*, y^*, s^*\right) = 0.$$

**Proof.** Substituting $(x^*, y^*, s^*)$ into (2.39) and (2.40), we have $r\left(x^*, y^*, s^*\right) = w\left(x^*, y^*, s^*\right) = 0$, which implies $\rho\left(x^*, y^*, s^*\right) = 0$. □

As mentioned at the beginning of this section, the identification function in [35] converges to zero when approaching the solution set, at a slower rate than the Euclidean distance. In the following proposition, we show that $\rho(x, y, s)$ behaves similarly.

**Proposition 3.2.** *Assume (PD) has a unique solution $(x^*, y^*, s^*)$ and let $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ be any point away from $(x^*, y^*, s^*)$, where $s = c - A^T y$. Then*

$$\lim_{(x,y,s)\to(x^*,y^*,s^*)} \frac{\rho(x, y, s)}{\|(x, y, s) - (x^*, y^*, s^*)\|} = +\infty, \tag{3.2}$$

*where $\|\cdot\|$ denotes the Euclidean norm.*

**Proof.** From Theorem 2.16 and norm properties, we have

$$\frac{\rho(x, y, s)}{\|(x, y, s) - (x^*, y^*, s^*)\|} \geq \frac{\rho(x, y, s)}{\|x - x^*\| + \|y - y^*\| + \|s - s^*\|} \geq \frac{\rho(x, y, s)}{\bar{\tau}\rho^2(x, y, s)} = \frac{1}{\bar{\tau}\rho(x, y, s)},$$

where $\bar{\tau} = \tau_p + \tau_y + \tau_d$. Thus

$$\lim_{(x,y,s)\to(x^*,y^*,s^*)} \frac{\rho(x, y, s)}{\|(x, y, s) - (x^*, y^*, s^*)\|} \geq \lim_{(x,y,s)\to(x^*,y^*,s^*)} \frac{1}{\bar{\tau}\rho(x, y, s)}.$$

This, the continuity of $\rho(x, y, s)$ and Proposition 3.1 imply that (3.2) holds. □

**Active-set prediction using the identification function.** Let $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ be an arbitrary point away from the (PD) solution set $\Omega$, and denote

$$\hat{\mathcal{A}}(x, y, s) = \{i \in \{1, \ldots, n\} |\ x_i \leq \rho(x, y, s)\}, \tag{3.3}$$

as the trial/predicted active set, and

$$\hat{\mathcal{A}}^+(x, y, s) = \{i \in \{1, \ldots, n\} |\ s_i \geq \rho(x, y, s)\}, \tag{3.4}$$

the trial/predicted strongly active set.

Next we show that the identification function (3.1) can indeed be used to identify the optimal active and strongly active sets under certain conditions. First we show that the distance from a triple $(x, y, s)$ to some optimal solution $(x^*, y^*, s^*)$ of (PD) is bounded by $\rho(x, y, s)$.

**Lemma 3.3.** *Let $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$, where $s = c - A^T y$. Then there exists a solution $(x^*, y^*, s^*)$ of (PD) such that*

$$\|x - x^*\| \leq \tau_p \rho^2(x, y, s) \quad \text{and} \quad \|s - s^*\| \leq \tau_d \rho^2(x, y, s),$$

*where $\tau_p$ and $\tau_d$ are problem dependent constants in (2.38), independent of $(x, y, s)$ and $(x^*, y^*, s^*)$.*

**Proof.** It follows directly from Theorem 2.16 and the definition of $\rho(x, y, s)$ in (3.1). □

**Theorem 3.4.** *Assume (PD) has a unique solution $(x^*, y^*, s^*)$. Let $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$, where $s = c - A^T y$. Then there exists a constant $\epsilon(x^*) > 0$ such that if $\rho(x, y, s) \leq \epsilon(x^*)$, then*

$$\hat{\mathcal{A}}(x, y, s) = \mathcal{A}(x^*),$$

*where $\hat{\mathcal{A}}(x, y, s)$ is defined in (3.3) and $\mathcal{A}(x^*)$ in (2.6a).*

**Proof.** From Lemma 3.3, $(x^*, y^*, s^*)$ being the unique solution of (PD) and norm properties, we have

$$|x_i - x_i^*| \leq \tau_p \rho^2(x, y, s), \quad \text{for all} \quad i \in \{1, \ldots, n\}. \tag{3.5}$$

If $i \in \mathcal{A}(x^*)$, $x_i^* = 0$. Assume $\rho(x, y, s) \leq \frac{1}{\tau_p}$. This and (3.5) give us that

$$x_i \leq x_i^* + \tau_p \rho^2(x, y, s) = \tau_p \rho^2(x, y, s) \leq \rho(x, y, s),$$

which implies $i \in \hat{\mathcal{A}}(x, y, s)$ and so $\mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x, y, s)$. If $i \notin \mathcal{A}(x^*)$, $x_i^* > 0$. From (3.5), we have $x_i \geq x_i^* - \tau_p \rho^2(x, y, s)$ and so $x_i > \rho(x, y, s)$ when $E(\rho) \equiv \tau_p \rho^2(x, y, s) + \rho(x, y, s) - x_i^* < 0$. Since $x_i^* > 0$, $E(\rho) < 0$ is satisfied for $0 < \rho(x, y, s) < t(x^*)$ where $t(x^*)$ is the positive root of the equation $E(\rho) = 0$. This implies $\hat{\mathcal{A}}(x, y, s) \subseteq \mathcal{A}(x^*)$ if $\rho(x, y, s) < t(x^*)$. Setting $\epsilon(x^*) = \min\left(\frac{1}{\tau_p}, t(x^*)\right)$, the result follows. □

**Theorem 3.5.** *Assume (PD) has a unique solution $(x^*, y^*, s^*)$. Let $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$, where $s = c - A^T y$. Then there exists a constant $\epsilon(s^*) > 0$ such that if $\rho(x, y, s) \leq \epsilon(s^*)$, then*

$$\hat{\mathcal{A}}^+(x, y, s) = \mathcal{A}^+(s^*),$$

*where $\hat{\mathcal{A}}^+(x,y,s)$ is defined in (3.3) and $\mathcal{A}^+(s^*)$ in (2.6b).*

**Proof.** From Lemma 3.3, $(x^*, y^*, s^*)$ being the unique solution of (PD), and norm properties, we have

$$|s_i - s_i^*| \leq \tau_d \rho^2(x,y,s), \quad \text{for all} \quad i \in \{1,\ldots,n\}. \tag{3.6}$$

If $i \in \mathcal{A}^+(s^*)$, $s_i^* > 0$. From (3.6), $s_i \geq s_i^* - \tau_d\rho^2(x,y,s)$ and so $s_i \geq \rho(x,y,s)$ when $\tau_d\rho^2(x,y,s) + \rho(x,y,s) - s_i^* \leq 0$. Thus, since $s_i^* > 0$, there exists some $t(s^*) > 0$ such that $s_i \geq \rho(x,y,s)$ when $\rho(x,y,s) < t(s^*)$. It follows that $\mathcal{A}^+(s^*) \subseteq \hat{\mathcal{A}}^+(x,y,s)$ if $\rho(x,y,s) < t(s^*)$. Suppose $i \notin \mathcal{A}^+(s^*)$, which implies $s_i^* = 0$. Let $\rho(x,y,s) \leq \frac{1}{\tau_d}$. This and (3.6) give us

$$s_i \leq s_i^* + \tau_d\rho^2(x,y,s) \leq \rho(x,y,s),$$

which implies $i \notin \hat{\mathcal{A}}^+(x,y,s)$ and so $\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*)$. The result follows by setting $\epsilon(s^*) = \min\left(\frac{1}{\tau_d}, t(s^*)\right)$. □

**Remark on the conditions in Theorems 3.4 and 3.5.** In Theorems 3.4 and 3.5, we require the (PD) problems have a unique primal-dual solution $(x^*, y^*, s^*)$. Then, note that $\rho(x,y,s) \to 0$ if and only if $(x,y,s) \to (x^*, y^*, s^*)$. (The forward implication follows from Proposition 3.1 and $\rho(x,y,s)$ continuous while the backward one from Theorem 2.16 and $(x^*, y^*, s^*)$ unique.) When multiple solutions are present, then (3.5) and (3.6) may include different solutions $(x^*, y^*, s^*)$ of (PD). Then, we can still prove, in the same way as above, that $\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A} \subseteq \hat{\mathcal{A}}(x,y,s)$, if $\rho(x,y,s)$ sufficiently small, where $(\mathcal{A}, \mathcal{I})$ is the strictly complementary partition. Note that in the numerical tests (Section 3.4), we do not assume or impose uniqueness of solution, and the test problems generally have multiple solutions.

Similar (but not identical) results for active-set prediction using the identification function for (PD) can be proved by setting the perturbations in Theorems 5.21 – 5.23 to zero.

## 3.2 Indicators

The term 'indicator' or 'indicator function' denote a function that can be used to identify the optimal active set of a constrained-problem. El-Bakry et al. [32] have studied and compared various indicators proposed in the literature (for example [124, 45, 69, 122, 84]), especially those that can be used in conjunction with primal-dual IPMs (Algorithm 2.1). The following material is mainly summarised from [32, Sections 3, 5 and 6].

Consider the indicators defined in the context of primal-dual IPMs. We update the iterates $(x^k, y^k, s^k)$ as described in Algorithm 2.1, namely, $(x^{k+1}, y^{k+1}, s^{k+1}) = (x^k, y^k, s^k) + \alpha^k(\Delta x^k, \Delta y^k, \Delta s^k)$, where $(\Delta x^k, \Delta y^k, \Delta s^k)$ is the Newton direction and $\alpha^k$ stands for the stepsize at iteration $k$. Let

$$z^k = (x^k, y^k, s^k) \quad \text{and} \quad \Delta z^k = (\Delta x^k, \Delta y^k, \Delta s^k).$$

The indicator function $I^c$ is a function of $z^k$ and $\Delta z^k$ which satisfies the property that if $\lim_{k\to\infty} z^k = (x^*, y^*, s^*)$, where $(x^*, y^*, s^*)$ is an optimal solution of (PD), then

$$\lim_{k\to\infty} I_i^c(z^k, \Delta z^k) = \begin{cases} \beta_i & \text{if } i \in \mathcal{A}(x^*) \\ \kappa_i & \text{if } i \notin \mathcal{A}(x^*) \end{cases}, \quad \text{for all} \quad i = 1, \ldots, n, \tag{3.7}$$

where $\beta_i$ and $\kappa_i$ are some constants satisfying $\max_{i\in\mathcal{A}(x^*)} \beta_i < \min_{i\notin\mathcal{A}(x^*)} \kappa_i$, and $\mathcal{A}(x^*)$ is the active set at $x^*$ defined in (2.6a). A larger gap between $\max_{i\in\mathcal{A}(x^*)} \beta_i$ and $\min_{i\notin\mathcal{A}(x^*)} \kappa_i$ implies that the given indicator could be more effective ('sharp separation' property [32]); see [32, Section 3] for some other properties that an indicator should ideally satisfy.

Indicators are incorporated inside a threshold test to determine the active constraints; $x_i^*$ is determined to be zero if $I_i^c$ is less than a user-defined threshold, namely

$$I_i^c(z^k, \Delta z^k) \leq \text{ threshold } \implies x_i^* = 0.$$

Next we introduce two types of indicators discussed in [32] and considered them to be the best. We compare the performance of the identification function defined in (3.1) with these indicators in Section 3.4.4.

**Primal-dual indicator.** Define the following indicator function

$$I^c(z^k, \Delta z^k, \alpha^k) = (S^{k+1})^{-1} X^{k+1} e, \tag{3.8}$$

where $S^{k+1} = \text{diag}(s^{k+1})$, $X^{k+1} = \text{diag}(x^{k+1})$, e is a vector of ones, and $z^{k+1} = z^k + \alpha^k \Delta z^k$. This indicator is known as the *primal-dual indicator*. For theoretical properties of this indicator in the context of primal-dual IPMs, see [32, Section 5.1].

**Remark.** According to [32, Proposition 7.2], under certain conditions on Algorithm 2.1, such as $(x^0, y^0, s^0)$ being strictly feasible, $(x^k)^T s^k \to 0$, $\frac{\min(X^k S^k e)}{(x^k)^T s^k}$ bounded below away from zero, and other conditions on algorithm parameters, we have that $I_i^c(z^k, \Delta z^k, \alpha^k)$ converges to zero $Q$-superlinearly for all $i \in \mathcal{A}(x^*)$. This is equivalent to a partial prediction result, namely for any user-defined positive threshold, when $k$ is sufficiently large, $\mathcal{A}(x^*) \subseteq \{i \in \{1, \ldots, n\} \mid I_i^c(z^k, \Delta z^k, \alpha^k) \leq \text{threshold}\}$. To the best of our knowledge, the converse inclusion has not been proven.

**Tapia indicators.** Tapia indicators consist of two indicator functions for the primal and dual problems in (PD), respectively. Namely,

$$I_p^c(z^k, \Delta z^k, \alpha^k) = (X^k)^{-1} X^{k+1} e \quad \text{and} \quad I_d^c(z^k, \Delta z^k, \alpha^k) = e - (S^k)^{-1} S^{k+1} e. \tag{3.9}$$

The properties of the above two indicators are discussed in detail in [32, Section 6.1].

**Remark.** Under the same conditions on Algorithm 2.1 as for the primal-dual indicator, [32, Proposition 6.1] shows that

$$(I_p^c(z^k, \Delta z^k, \alpha^k))_i \to \begin{cases} 0 & i \in \mathcal{A}(x^*) \\ 1 & i \notin \mathcal{A}(x^*) \end{cases} \quad \text{and} \quad (I_d^c(z^k, \Delta z^k, \alpha^k))_i \to \begin{cases} 0 & i \in \mathcal{A}(x^*) \\ 1 & i \notin \mathcal{A}(x^*) \end{cases}.$$

Letting the threshold be between (0,1), this result can be easily transformed into a prediction result, namely, for sufficiently large $k$, $\{i \in \{1, \ldots, n\} \,|\, (I_p^c(z^k, \Delta z^k, \alpha^k))_i \leq \text{threshold}\}$ and $\{i \in \{1, \ldots, n\} \,|\, (I_d^c(z^k, \Delta z^k, \alpha^k))_i \leq \text{threshold}\}$ are both equivalent to $\mathcal{A}(x^*)$. Under further conditions on algorithm parameters, Proposition 7.3 in [32] gives that $I_p^c(z^k, \Delta z^k, \alpha^k)$ converges with an $R$-rate of convergence between $(1, 2]$. A similar result can be obtained for $I_d^c(z^k, \Delta z^k, \alpha^k)$ using both Propositions 7.3 and 7.4 in [32].

## 3.3 Simple cut-off

The term 'cut-off' stands for the simplest way of performing active-set prediction, which splits the variables into active or inactive based on whether they are less than a user-defined small value, namely,

$$x_i^k \leq \text{threshold} \implies x_i^* = 0.$$

This can also be viewed as "using variables as indicators" [32, Section 4]. This strategy is widely used in our theoretical and numerical approaches in Chapters 5 – 7, due essentially to its simplicity.

Under certain conditions on problem conditioning (see Theorem 5.7 with $\lambda = 0$), the predicted active set coincides with the original (PD) optimal active set when the duality gap is sufficiently small. This result can be proved by setting the perturbations in Theorem 5.7 to zero.

## 3.4 Numerical experiments

### 3.4.1 Implementation

**Primal-dual path-following IPM.** All tests in this chapter are conducted in the context of the infeasible primal-dual path-following IPM (Algorihm 2.1, Section 2.2.2) with the starting point suggested by Mehrotra [93] (see page 22) and the active-set

prediction step incorporated into it; for details, see Algorithm 3.1 and the explanations thereafter.

**Algorithm 3.1** A Primal-Dual Path-Following IPM with Active Set Prediction

**Step 0**: Calculate a starting point $(x^0, y^0, s^0)$ with $(x^0, s^0) > 0$ for (PD) following Mehrotra's procedure [93] (given on page 22);

**for** $k = 0, 1, 2, \ldots$ **do**

**Step 1:** solve the system (2.12) by the augmented system approach (2.14) to obtain the Newton direction $(\Delta x^k, \Delta y^k, \Delta s^k)$;

**Step 2:** compute possibly distinct stepsizes $\alpha_p^k$ for the primal iterates $\Delta x^k$ and $\alpha_d^k$ for the dual ones $(\Delta y^k, \Delta s^k)$ following (2.17);

**Step 3:** update $x^{k+1} = x^k + \alpha_p^k \Delta x^k$ and $(y^{k+1}, s^{k+1}) = (y^k, s^k) + \alpha_d^k (\Delta y^k, \Delta s^k)$;

**Step 4:** predict the optimal active set of (PD) and denote it by $\mathcal{A}^k$;

**Step 5:** terminate if some termination criterion is satisfied;

**end for**

**Active-set prediction framework.** In our numerical test, we apply a more complex strategy to predict the active constraints, inspired by [32, Step 3 in Procedure 8.1]. We partition the index set $\{1, 2, \ldots, n\}$ into three sets, $\mathcal{A}^k$ as the predicted active set, $\mathcal{I}^k$ as the predicted inactive set and $\mathcal{Z}^k = \{1, 2, \ldots, n\} \backslash \left(\mathcal{A}^k \cup \mathcal{I}^k\right)$ which includes all undetermined indices. During the running of the algorithm, we move indices between these sets according to certain **threshold tests**.

Initialise $\mathcal{A}^0 = \mathcal{I}^0 = \emptyset$ and $\mathcal{Z}^0 = \{1, 2, \ldots, n\}$. An index is moved from $\mathcal{Z}^k$ to $\mathcal{A}^k$ if the threshold test is satisfied for two consecutive iterations, otherwise from $\mathcal{Z}^k$ to $\mathcal{I}^k$. We move an index from $\mathcal{A}^k$ to $\mathcal{Z}^k$ if the threshold test is not satisfied at the current iteration. An index is moved from $\mathcal{I}^k$ to $\mathcal{Z}^k$ if the threshold test is satisfied at the current iteration. We summarise the above as Procedure 3.2.

**Identification function.** We use both primal and dual information to conduct the threshold test, namely

$$x_i^k < \rho(x^k, y^k, s^k) \quad \text{and} \quad s_i^k > \rho(x^k, y^k, s^k), \quad i \in \{1, 2, \ldots, n\}.$$

**Indicators.** We follow [32] and utilise a combination of the primal-dual indicator and the Tapia indicators. Assume $IPD_i^k$ is the value of the primal-dual indicator and $IT_i^k$ the value of the sum of two Tapia indicators at iteration $k$ respectively, namely

$$IPD_i^k = \left| \frac{x_i^{k+1}}{s_i^{k+1}} \right| \quad \text{and} \quad IT_i^k = \left| \frac{x_i^{k+1}}{x_i^k} \right| + \left| 1 - \frac{s_i^{k+1}}{s_i^k} \right|, \quad \forall i \in \{1, 2, \ldots, n\}.$$

As suggested in [32], we choose 0.1 as the threshold for the primal-dual indicator and 0.2 for the sum of the Tapia indicators. So the threshold test for indicators is defined

**Procedure 3.2** An Active-set Prediction Procedure

**Initialise**: $A^0 = \mathcal{I}^0 = \emptyset$ and $\mathcal{Z}^0 = \{1, 2, \ldots, n\}$.
**At** $k^{th}$ **iteration,** $k > 1$,
**for** $i = 1, \ldots, n$ **do**
  **if** $i \in \mathcal{Z}^k$ **then**
    **if** the threshold test is satisfied for iterations $k-1$ and $k$ **then**
      $\mathcal{A}^k = \mathcal{A}^k \cup \{i\}$ and $\mathcal{Z}^k = \mathcal{Z}^k \backslash \{i\}$;
    **else**
      $\mathcal{I}^k = \mathcal{I}^k \cup \{i\}$ and $\mathcal{Z}^k = \mathcal{Z}^k \backslash \{i\}$.
    **end if**
    **if** $i \in \mathcal{A}^k$ and the threshold test is not satisfied **then**
      $\mathcal{A}^k = \mathcal{A}^k \backslash \{i\}$ and $\mathcal{Z}^k = \mathcal{Z}^k \cup \{i\}$;
    **end if**
    **if** $i \in \mathcal{I}^k$ and the threshold test is satisfied **then**
      $\mathcal{I}^k = \mathcal{I}^k \backslash \{i\}$ and $\mathcal{Z}^k = \mathcal{Z}^k \cup \{i\}$.
    **end if**
  **end if**
**end for**

as

$$IPD_i^k < 0.1 \quad \text{and} \quad IT_i^k < 0.2, \quad i \in \{1, 2, \ldots, n\}.$$

**Cut-off.** We employ the following simple test

$$x_i^k < 10^{-5} \quad \text{and} \quad s_i^k > 10^{-5}, \quad i \in \{1, 2, \ldots, n\}.$$

**Preprocessing.** A common assumption for IPM algorithms is that the matrix $A$ needs to have full row rank (Assumption (2.9)). This is not a stringent requirement as a matrix can always be reduced to a full row rank matrix [135, Page 31-32] without too much computational effort. In our tests, we apply the preprocessing code from LIPSOL [140] to ensure this condition.

### 3.4.2 Test problems

**Randomly generated test problems (TS1).** We first randomly generate the number of constraints $m \in (10, 200)$, the number of variables $n \in (20, 500)$ and density of nonzero entries in $A$ within $(0.4, 0.8)$, where $m < n$, $2m < n < 7m$. Then randomly generate a matrix $A \in \mathbb{R}^{m \times n}$ of given density and a point $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ with $x \geq 0$, $s \geq 0$ and density about 0.5. Finally we generate $b$ and $c$ by letting $b = Ax$ and $c = A^\top y + s$. Thus $(x, y, s)$ serves as a feasible point. Problems generated this way are generally well-conditioned and primal nondegenerate. This test set is inspired by the random problem generation approach in [36, Section 8.3.4 ]. Whenever we use this test set, (the same) 100 problems are generated.

We checked the degeneracy of 100 test problems generated this way, by looking at the vertex solutions obtained by the MATLAB simplex solver. The majority of these

problems are primal nondegenerate and dual degenerate.

**Randomly generated primal-dual degenerate test problems (TS2).** Instead of generating a feasible point as for TS1, we generate $(x, y, s)$ with $x \geq 0$, $s \geq 0$, $x_i s_i = 0$ for all $i \in \{1, \ldots, n\}$ so that the number of nonzeros of $x$ is strictly less than $m$ and that of $s$ is strictly less than $n - m$. Then get $A$, $b$, $c$ as for TS1. Thus $(x, y, s)$ serves as a primal-dual degenerate solution. 100 problems are also generated for this test set and used for all tests.

**Netlib problems (TS3).** Netlib [44] is a collection of standard LP test problems. The original test problems can be found at http://www.netlib.org/lp/data/.[6]

Most Netlib test problems are not in the standard form. We reformulate them into the standard form by introducing slacks. Since our implementation is basic, in MATLAB, and mainly for illustration, we choose a subset of problems in Netlib with the number of primal variables less than 5000 (including the slack variables). See Table 3.1 for the list of the 37 Netlib problems selected.

Table 3.1: Selected 37 Netlib problems

| Name | $m$ | $n$ | Name | $m$ | $n$ |
|---|---|---|---|---|---|
| 25FV47 | 798 | 1854 | ADLITTLE | 55 | 137 |
| AFIRO | 27 | 51 | AGG3 | 516 | 758 |
| BLEND | 74 | 114 | BNL1 | 632 | 1576 |
| BRANDY | 149 | 259 | CZPROB | 737 | 3141 |
| E226 | 220 | 469 | FIT1D | 1050 | 2075 |
| FIT1P | 1026 | 2076 | FORPLAN | 157 | 485 |
| GROW7 | 420 | 581 | ISRAEL | 174 | 316 |
| KB2 | 52 | 77 | SC50A | 49 | 77 |
| SC50B | 48 | 76 | SCAGR7 | 129 | 185 |
| SCFXM1 | 322 | 592 | SCFXM2 | 644 | 1184 |
| SCFXM3 | 966 | 1776 | SCRS8 | 485 | 1270 |
| SCSD1 | 77 | 760 | SCSD6 | 147 | 1350 |
| SCSD8 | 397 | 2750 | SCTAP1 | 300 | 660 |
| SCTAP2 | 1090 | 2500 | SCTAP3 | 1480 | 3340 |
| SEBA | 1029 | 1550 | SHARE1B | 112 | 248 |
| SHARE2B | 96 | 162 | SHIP04L | 356 | 2162 |
| SHIP08L | 688 | 4339 | SHIP08S | 416 | 2171 |
| SHIP12S | 466 | 2293 | STAIR | 362 | 544 |
| STOCFOR2 | 2157 | 3045 | | | |

In addition, six of these problems, AFIOR, ADLITTLE, SCSD1, SHIP04L, SHARE2B, and GROW7, have also been tested in [32, Section 8] in the aim of analysing the behaviour of different indicators.

[6] We use a MATLAB version of the same test set, which is obtained from http://www.math.ntu.edu.tw/~wwang/cola_lab/test_problems/netlib_lp/. Additionally, the reader can also download the full set of test problems in .mat format from my open source collection on Github https://github.com/YimingYAN/LP-Test-Problems.

### 3.4.3 Prediction ratios

Assume $\mathcal{A}^k$ is the predicted active set at iteration $k$ and $\mathcal{A}$ is the actual optimal active set. To compare the accuracy of the predictions, we introduce the following three prediction ratios.

- False-prediction ratio $= \frac{|\mathcal{A}^k \setminus (\mathcal{A}^k \cap \mathcal{A})|}{|\mathcal{A}^k \cup \mathcal{A}|}$.
- Missed-prediction ratio $= \frac{|\mathcal{A} \setminus (\mathcal{A}^k \cap \mathcal{A})|}{|\mathcal{A}^k \cup \mathcal{A}|}$.
- Correctness ratio $= \frac{|\mathcal{A}^k \cap \mathcal{A}|}{|\mathcal{A}^k \cup \mathcal{A}|}$.

False-prediction ratio measures the degree of incorrectly identified active constraints, missed-prediction ratio measures the degree of incorrectly rejected active constraints and correctness ratio shows the accuracy of the prediction. All three ratios range from 0 to 1. If the predicted set is the same as the actual optimal active set, correctness ratio is 1. See Figure 3.1 for an illustration. These ratios are employed to compare the accuracy of active-set predictions in our numerical tests throughout this thesis.

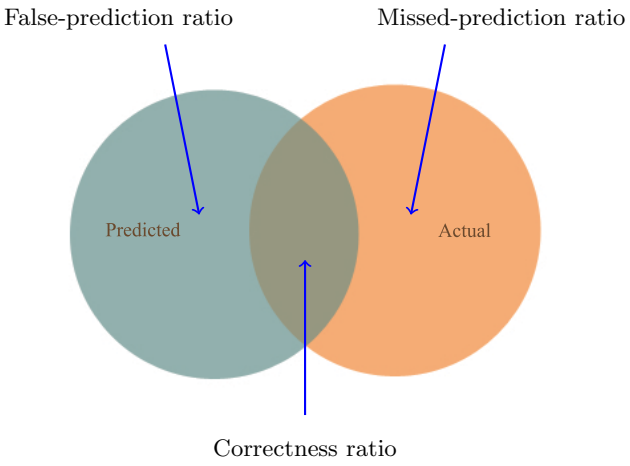



Figure 3.1: An illustration of prediction ratios

### 3.4.4 Numerical results

**Randomly generated problems (TS1 and TS2).** In this test, we aim to compare the accuracy of active-set predictions using the identification function, indicators and cut-off. We first obtain the 'actual optimal active set' by solving the test problem using MATLAB's solver LINPROG with the 'algorithm' option set to 'interior point' and considering all variables less than $10^{-5}$ as active.[7] Then we run the Algorithm 3.1, terminate the algorithm at each iteration, predict the active set using Procedure 3.2, and compare the predicted active set with the actual optimal active set.

[7] We use the default termination tolerance ($10^{-8}$) for interior point solver in LINPROG, and we consider it is accurate enough for our purpose of tests.

In Figures 3.2 and 3.3, we present the results for TS1 (left) and TS2 (right). The x-axis shows the number of interior point iterations at which we terminate Algorithm 3.1. In each figure, the first three plots (from left to right, top to bottom) show the average value of the three measures mentioned above for the test problems in question. The last plot at the bottom right corner presents the corresponding $\log_{10}$ scaled relative KKT residuals defined in (2.18). There are three lines in each plot, representing the prediction ratios by comparing the active set predicted by the identification function with the actual active set (solid red line with circle), that by indicators (dashed blue line with star) and that by cut-off (dashed black line with square) respectively.

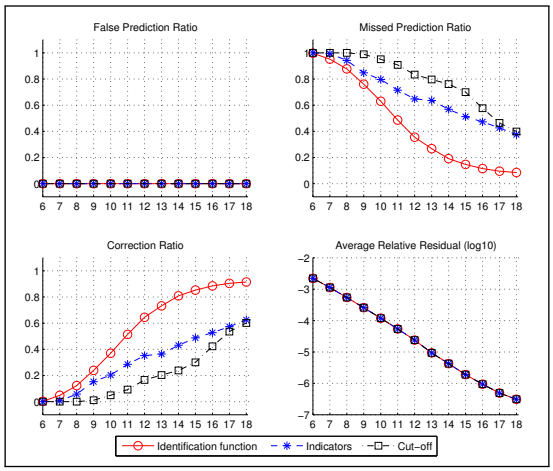



Figure 3.2: Comparing prediction ratios for the identification function, indicators and cut-off on randomly generated problems

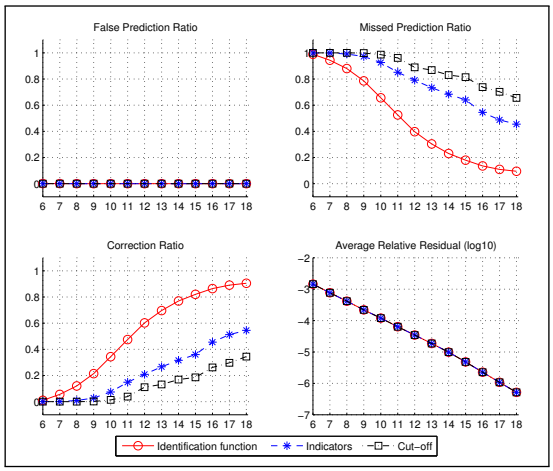



Figure 3.3: Comparing prediction ratios for the identification function, indicators and cut-off on randomly generated primal-dual degenerate problems

- Figures 3.2 and 3.3 show that the average correctness ratios for the identification function are at least as good and generally better — namely, more than 4 times higher at certain iterations — than the indicators and cut-off.
- The performance of indicators is generally better — namely, about 2 times higher at certain iterations — than that of cut-off in the context of correctness ratios. The gap, however, is not as much as that between the identification function and the indicators.
- At the $10^{\text{th}}$ iteration, when the average relative residual is about $10^{-4}$, none of the three methods can predict more than 40% of the active set. It seems that none of them are very efficient for predicting the active set early on.
- After 18 iterations, the correctness ratios do not reach 1. This is due to ill-conditioning which prevents us from solving any further.

NETLIB **problems (TS3).** In this part, we compare the prediction ratios on 37 NETLIB problems. For each test problem, we first solve it to optimality (relative residual

in (2.18) $< 10^{-6}$) using Algorithm 3.1 and record the total number of iterations needed, say $M$. Then we compare the average prediction ratios at the last 6 iterations of each test case. This is because the number of iterations needed for each NETLIB test problem varies too much, and so it is not appropriate to compare at some fixed iterations.[8]

In Figure 3.4, we observe similar phenomena as for the random tests. For example, when the average relative residual is about $10^{-4}$, the average correctness ratio for the identification function is the highest, about 43% at $M-3$ and 50% at $M-2$; for indicators and cut-off, they are only about 5%.

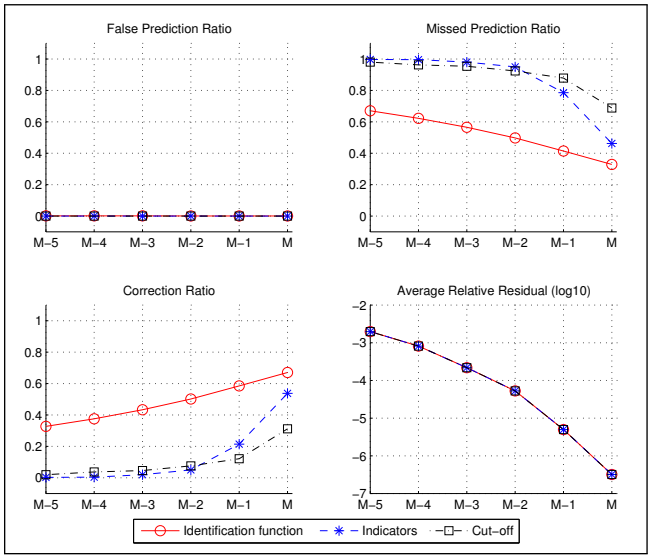



Figure 3.4: Comparing prediction ratios for the identification function, indicators and cut-off on 6 NETLIB problems

**Remarks on the choice of cut-off.**

- We may find an accurate cut-off for each problem if we solve the problem to a greater accuracy so that the zero and nonzero variables are well separated. However this is not realistic, because in general it is not known a priori to what accuracy a particular problem should be solved and ill-conditioning can prevent the solver from getting to a desired accuracy [32]. In the literature, small values of cut-off are suggested, for example, $10^{-6}$ is used in [45], and in [89] they adopt the accuracy of their algorithm as the value of cut-off, which is $10^{-8}$. Considering that in our tests in general we solve the test problems to the accuracy of $\mathcal{O}(10^{-6})$ and the active-set prediction procedure (Procedure 3.1) also uses the dual information as a safeguard, we choose $10^{-5}$ as the value of cut-off.

- For these particular test sets, we can increase the value of cut-off from $10^{-5}$ to $10^{-3}$ without affecting the false-prediction ratios too much. Though the performance of cut-off can be improved, it is not better than the identification function

[8] We have also conducted this test on the random problems and the results are generally the same as those in Figures 3.2 and 3.3.

and still can only predict less than 40% of the active set at early stages of the iterative process (when the relative residual is about $10^{-4}$).

## 3.5 Conclusions

From the above numerical tests, it seems that the identification function works well, at least the same and generally better than indicators and cut-off when used in an infeasible primal-dual path-following interior point method. However, none of them are ideal for predicting the active set early on. Is it possible to have a satisfying prediction of the active set early on, say for example being able to predict over 70% of the active constraints before the relative residual (or duality gap) is less than $10^{-4}$? The main focus of the rest of this thesis is to tackle this question. From the next chapter, we start to derive a method utilising the so-called controlled perturbations to overcome this difficulty. We will show that even with the simplest prediction method, the cut-off, we often have some ability to predict sooner and better.

# 4 Perturbed Linear Programming Problems

The original idea of controlled perturbations was introduced by Cartis and Gould [24], in the context of finding 'well-centred' feasible points for LP. Inspired by their idea, we enlarge the feasible set of the original primal-dual problems (PD) by using controlled perturbations to formulate the perturbed variants of the (PD) problems and present some useful properties. In Section 4.1, we introduce the perturbed problems. In Section 4.2.1, we first show that the optimal solution of the original primal-dual problems lies on or in a neighbourhood of the central path of the perturbed primal-dual problems for carefully chosen perturbations. In Section 4.2.2, we find conditions such that the active set of the perturbed (primal) problem is the same as that of the original (primal) problem.

## 4.1 Controlled perturbations for linear programming

Consider the pair of primal-dual LP problems in (PD) (see page 16). We enlarge the feasible set of (PD) by using *controlled perturbations* [24], namely, we relax the nonnegativity constraints in (PD) and consider the pair of perturbed problems,

$$\begin{array}{lll} \text{(Primal)} & & \\ \min\limits_{x\in\mathbb{R}^n} & (c+\lambda)^T(x+\lambda) & \\ \text{s.t.} & Ax = b, & \\ & x \geq -\lambda, & \end{array} \qquad \begin{array}{lll} \text{(Dual)} & & \\ \max\limits_{(y,s)\in\mathbb{R}^m\times\mathbb{R}^n} & (b+A\lambda)^T y & \\ \text{s.t.} & A^T y + s = c, & \\ & s \geq -\lambda, & \end{array} \tag{PD$_\lambda$}$$

for some vector of perturbations $\lambda = (\lambda_1, \ldots, \lambda_n) \geq 0$. Different perturbations for $x$ and $s$ could be used, but for simplicity, we use the same vector of perturbations for both. To make these problems primal and dual to each other, perturbations have to be added to both the primal and dual objectives. Note that if $\lambda$ is a vector of zeros, (PD$_\lambda$) coincides with (PD). In Proposition 4.1 below we show the problems in (PD$_\lambda$) are indeed dual to each other.

Similarly to the strictly feasible set $\mathcal{F}^0$ of (PD), defined in (2.2), we denote the set

of strictly feasible points of (PD$_\lambda$),

$$\mathcal{F}_\lambda^0 = \left\{(x, y, s) \,\middle|\, Ax = b,\ A^T y + s = c,\ x + \lambda > 0,\ s + \lambda > 0\right\}. \tag{4.1}$$

Writing down the first order optimality conditions (KKT conditions) for (PD$_\lambda$), according for example to [105, Theorem 12.1], we find that $(x_\lambda^*, y_\lambda^*, s_\lambda^*)$ is a *(primal-dual) solution* for (PD$_\lambda$) if and only if it satisfies the following system,

$$Ax = b, \tag{4.2a}$$
$$A^T y + s = c, \tag{4.2b}$$
$$(X + \Lambda)(S + \Lambda)e = 0, \tag{4.2c}$$
$$(x + \lambda, s + \lambda) \geq 0, \tag{4.2d}$$

where $\Lambda = \operatorname{diag}(\lambda)$, $X = \operatorname{diag}(x)$, $S = \operatorname{diag}(s)$ and $e = (1, \ldots, 1)$. The solution $(x_\lambda^*, y_\lambda^*, s_\lambda^*)$ is a *(strictly complementary) solution* for (PD$_\lambda$) if

$$(x_\lambda^* + \lambda) + (s_\lambda^* + \lambda) > 0. \tag{4.3}$$

Denote by $\Omega_\lambda^P$ and $\Omega_\lambda^D$ the primal and dual solution sets of the (PD$_\lambda$) problems, respectively,

$$\begin{aligned} \Omega_\lambda^P &= \{x^* \,|\, x^* \text{ solves the primal problem in (PD}_\lambda\text{) }\}, \\ \Omega_\lambda^D &= \{(y^*, s^*) \,|\, (y^*, s^*) \text{ solves the dual problem in (PD}_\lambda\text{) }\}. \end{aligned} \tag{4.4}$$

Let $(x_\lambda^*, y_\lambda^*, s_\lambda^*)$ be a strictly complementary solution of (PD$_\lambda$), and denote

$$\mathcal{A}_\lambda = \{i \in \{1, \ldots, n\} | (x_\lambda^*)_i + \lambda_i = 0\} \quad \text{and} \quad \mathcal{I}_\lambda = \{i \in \{1, \ldots, n\} | (s_\lambda^*)_i + \lambda_i = 0\}. \tag{4.5}$$

(4.2c) and (4.3) imply that $\mathcal{A}_\lambda \cup \mathcal{I}_\lambda = \{1, \ldots, n\}$ and $\mathcal{A}_\lambda \cap \mathcal{I}_\lambda = \emptyset$, and so $(\mathcal{A}_\lambda, \mathcal{I}_\lambda)$ is the strictly complementary active and inactive partition of the solution set of (PD$_\lambda$).

**Equivalent formulation of (PD$_\lambda$).** In some cases, it is more convenient to have the equivalent 'standard form' of (PD$_\lambda$). Letting $p = x + \lambda$ and $q = s + \lambda$, we can write (PD$_\lambda$) in the following equivalent form,

$$\begin{array}{ll|ll} \text{(Primal)} & & \text{(Dual)} & \\ \min_p & c_\lambda^T p & \max_{(y,q)} & b_\lambda^T y \\ \text{s.t.} & Ap = b_\lambda, & \text{s.t.} & A^T y + q = c_\lambda, \\ & p \geq 0, & & q \geq 0, \end{array} \tag{4.6}$$

where $c_\lambda = c + \lambda$, $b_\lambda = b + A\lambda$ and $\lambda \geq 0$. The KKT conditions (Theorem 2.2) ensure that $(p^*_\lambda, y^*_\lambda, q^*_\lambda)$ is the (primal-dual) solution of (4.6) if and only if it satisfies

$$\begin{aligned} Ap &= b_\lambda, \\ A^T y + q &= c_\lambda, \\ PQe &= 0, \\ (p, q) &\geq 0, \end{aligned} \tag{4.7}$$

where $P = \mathrm{diag}(p)$, $Q = \mathrm{diag}(q)$ and $e = (1, \ldots, 1)$.

We now use (4.6) to show the duality of ($\mathrm{PD}_\lambda$). Note that the following proposition and its proof are inspired by [24, Lemma 5.1] and associated discussion on the problem formulation of the perturbed problems.

**Proposition 4.1.** *The pair of problems in ($\mathrm{PD}_\lambda$) are dual to each other.*

**Proof.** Letting $p := x + \lambda$ and $s := s + \lambda$, we reformulate the problems in ($\mathrm{PD}_\lambda$) to a pair of problems that are in the standard form (see (4.6)). It is easy to check that the problems in (4.6) are dual to each other since they are of the same format as problems (PD) . □

The following two propositions show that $(x^*_\lambda, y^*_\lambda, s^*_\lambda)$ is a ($\mathrm{PD}_\lambda$) solution if and only if $(p^*_\lambda, y^*_\lambda, q^*_\lambda)$, where $p^*_\lambda = x^*_\lambda + \lambda$ and $q^*_\lambda = s^*_\lambda + \lambda$, is a solution of (4.6). Thus we can construct an optimal solution for ($\mathrm{PD}_\lambda$) from an optimal solution of (4.6) and vice versa.

**Proposition 4.2.** *If $(x^*_\lambda, y^*_\lambda, s^*_\lambda)$ is an optimal solution of ($\mathrm{PD}_\lambda$), then $(p^*_\lambda, y^*_\lambda, q^*_\lambda)$ is an optimal solution of (4.6), where $p^*_\lambda = x^*_\lambda + \lambda$ and $q^*_\lambda = s^*_\lambda + \lambda$.*

**Proof.** Since $(x^*_\lambda, y^*_\lambda, s^*_\lambda)$ is an optimal solution of ($\mathrm{PD}_\lambda$), $p^*_\lambda = x^*_\lambda + \lambda$ and $q^*_\lambda = s^*_\lambda + \lambda$, then from (4.1), we have

$$\begin{aligned} Ap^*_\lambda &= A(x^*_\lambda + \lambda) = b + A\lambda = b_\lambda, \\ A^T y^*_\lambda + q^*_\lambda &= A^T y^*_\lambda + s^*_\lambda + \lambda = c_\lambda, \\ P^*_\lambda Q^*_\lambda e &= (X^*_\lambda + \lambda)(S^*_\lambda + \Lambda)e = 0, \\ (p^*_\lambda, q^*_\lambda) &\geq 0, \end{aligned}$$

namely, the KKT conditions (4.7) are satisfied. □

**Proposition 4.3.** *If $(p^*_\lambda, y^*_\lambda, q^*_\lambda)$ is an optimal solution of (4.6), then $(x^*_\lambda, y^*_\lambda, s^*_\lambda)$ is an optimal solution of ($\mathrm{PD}_\lambda$), where $x^*_\lambda = p^*_\lambda - \lambda$ and $s^*_\lambda = q^*_\lambda - \lambda$.*

**Proof.** It is obvious that

$$\begin{aligned} Ax^*_\lambda &= A(p^*_\lambda - \lambda) = b_\lambda - A\lambda = b, \\ A^T y^*_\lambda + s^*_\lambda &= A^T y^*_\lambda + q^*_\lambda - \lambda = c_\lambda - \lambda = c, \\ (X^*_\lambda + \Lambda)(S^*_\lambda + \Lambda)e &= P^*_\lambda Q^*_\lambda e = 0, \\ (x^*_\lambda + \lambda, s^*_\lambda + \lambda) &= (p^*_\lambda, q^*_\lambda) \geq 0, \end{aligned}$$

namely, the KKT conditions (4.2) are satisfied. □

**The central path of (PD$_\lambda$).** Following [135, Chapter 2] or the procedure in Section 2.2.1, we formulate the logarithmic barrier subproblems of (PD$_\lambda$), express the Lagrangian of this subproblem and then derive its first order optimality conditions,

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ (X+\Lambda)(S+\Lambda)e &= \mu\, e, \\ (x+\lambda, s+\lambda) &> 0, \end{aligned} \tag{4.8}$$

where $\mu > 0$ is the barrier parameter for the perturbed problem (PD$_\lambda$). (4.8) describes the central path equations for (PD$_\lambda$). The central path of (PD$_\lambda$) is well defined under mild conditions.

**Lemma 4.4 (Existence and convergence of the perturbed central path).** *Let (2.9)[9] hold and $\lambda \geq 0$. Then the central path of the perturbed problems (PD$_\lambda$) is well defined, namely, the system (4.8) has a unique solution $(x_\lambda(\mu), y_\lambda(\mu), s_\lambda(\mu))$ for each $\mu > 0$, provided $\mathcal{F}_\lambda^0$ in (4.1) is nonempty. In particular, if $\lambda > 0$, $\mathcal{F}_\lambda^0$ is nonempty whenever (PD) has a nonempty primal-dual feasible set. Furthermore, $(x_\lambda(\mu), y_\lambda(\mu), s_\lambda(\mu))$ converges to a strictly complementary solution of (PD$_\lambda$), as $\mu \to 0$.*

The existence result is from of Lemma 5.1 in [24]. Considering the equivalent form (4.6) of (PD$_\lambda$), the convergence result follows from Theorem 2.7.

**Remark.** Note that if $\lambda > 0$, the condition required in Lemma 4.4 for the existence of the perturbed central path is weaker than that for the central path of (PD). The latter requires (PD) to have a nonempty *strictly* feasible set $\mathcal{F}^0$ in (2.2), namely, for there to be (PD) feasible points that strictly satisfy all problem inequality constraints (Theorem 2.6) while for (PD$_\lambda$), we only need $\mathcal{F}$, the feasible set of (PD) to be nonempty. When $\lambda = 0$, the existence and convergence results are equivalent to those for (PD), namely Theorems 2.6 and 2.7 respectively.

## 4.2 Perturbed problems and their properties

### 4.2.1 Perfect and relaxed perturbations

Geometrically, the original optimal solution $(x^*, y^*, s^*)$ of (PD) may lie on or near the central path of the perturbed problem (PD$_\lambda$) for carefully chosen perturbations; see Figures 4.1 and 4.2. Algebraically, this happens if $(x^*, y^*, s^*)$ satisfies the third relation in (4.8) exactly or approximately. We make these considerations precise in the next two theorems.

[9] Recall that assumption (2.9) is simply the full row rank assumption for the matrix $A$.

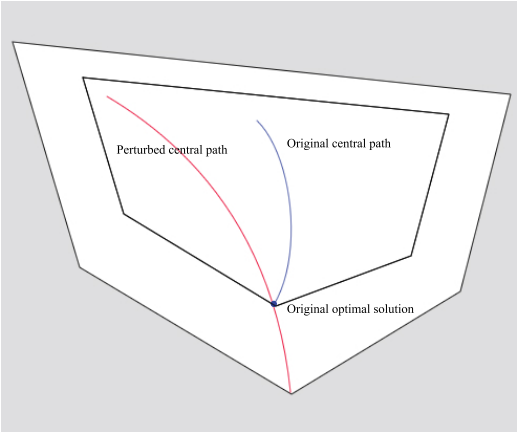



Figure 4.1: Perfect perturbations.

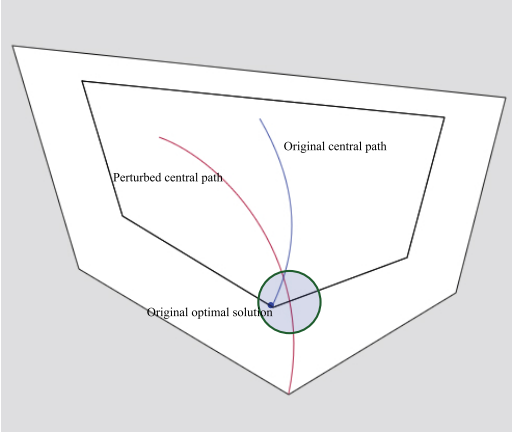



Figure 4.2: Relaxed perturbations.

**Theorem 4.5 (Existence of 'perfect' perturbations).** *Assume* (2.9) *holds and* $(x^*, y^*, s^*)$ *is a solution of* (PD). *Let* $\hat{\mu} > 0$. *Then there exists a vector of perturbations*

$$\hat{\lambda} = \hat{\lambda}(x^*, s^*, \hat{\mu}) > 0,$$

*such that the perturbed central path* (4.8) *with* $\lambda = \hat{\lambda}$ *passes through* $(x^*, y^*, s^*)$ *exactly when* $\mu = \hat{\mu}$.

**Proof.** Since $(x^*, y^*, s^*)$ is an optimal solution of (PD), it is also primal-dual feasible, and so $(x^*, y^*, s^*) \in \mathcal{F}_\lambda^0$ for any $\lambda > 0$. Thus, according to Lemma 4.4, the perturbed central path is well defined. Furthermore, if there exists a $\hat{\lambda} > 0$ such that

$$\left(X^* + \hat{\Lambda}\right)\left(S^* + \hat{\Lambda}\right)e = \hat{\mu}e, \tag{4.9}$$

then $(x^*, y^*, s^*)$ is the unique solution of the perturbed central path equations (4.8) with $\lambda = \hat{\lambda}$ and $\mu = \hat{\mu}$, which implies the central path of perturbed problems passes through $(x^*, y^*, s^*)$. It remains to solve (4.9) for $\hat{\lambda} = (\hat{\lambda}_1, \ldots, \hat{\lambda}_n)$. Since $x_i^* s_i^* = 0$, $i = 1, \ldots, n$, we have that (4.9) is equivalent to

$$\hat{\lambda}_i^2 + (x_i^* + s_i^*)\,\hat{\lambda}_i - \hat{\mu} = 0, \quad i = 1, \ldots, n,$$

whose positive root for each $i$ gives the corresponding component of the required $\hat{\lambda}$. □

It is a stringent and impractical requirement to force the optimal solution of the original problem to be exactly on the central path of the perturbed problems. Thus we relax this requirement to allow for the original solution to belong to a small neighbourhood of this path.

**Theorem 4.6 (Existence of relaxed perturbations).** *Assume* (2.9) *holds and* $(x^*, y^*, s^*)$ *is a* (PD) *solution and let* $\hat{\mu} > 0$ *and* $\xi \in (0,1)$. *Then there exist vetors* $\hat{\lambda}_L = \hat{\lambda}_L(x^*, s^*, \hat{\mu}, \xi) > 0$ *and* $\hat{\lambda}_U = \hat{\lambda}_U(x^*, s^*, \hat{\mu}, \xi) > 0$ *such that for* $\hat{\lambda}_L \leq \lambda \leq \hat{\lambda}_U$, $(x^*, y^*, s^*)$ *is strictly feasible for* ($\text{PD}_\lambda$) *and satisfies*

$$\xi\hat{\mu}e \leq (X^* + \Lambda)(S^* + \Lambda)e \leq \frac{1}{\xi}\hat{\mu}e. \tag{4.10}$$

**Proof.** Clearly, $(x^*, y^*, s^*)$ satisfies (4.1) and so $(x^*, y^*, s^*) \in \mathcal{F}_\lambda^0$ for any $\lambda = (\lambda_1, \ldots, \lambda_n) > 0$. The inequalities (4.10) are equivalent to

$$\begin{cases} \lambda_i^2 + (x_i^* + s_i^*)\lambda_i - \xi\hat{\mu} & \geq & 0 \\ \lambda_i^2 + (x_i^* + s_i^*)\lambda_i - \frac{1}{\xi}\hat{\mu} & \leq & 0, \end{cases} \tag{4.11}$$

for all $i \in \{1, \ldots, n\}$ and $\xi \in (0,1)$. Solving (4.11) for $\lambda_i$, we obtain

$$\begin{cases} \lambda_i \geq & \frac{-(x_i^*+s_i^*)+\sqrt{(x_i^*+s_i^*)^2+4\xi\hat{\mu}}}{2} = & \frac{2\xi\hat{\mu}}{x_i^*+s_i^*+\sqrt{(x_i^*+s_i^*)^2+4\xi\hat{\mu}}} = (\hat{\lambda}_L)_i, \\ 0 < \lambda_i \leq & \frac{-(x_i^*+s_i^*)+\sqrt{(x_i^*+s_i^*)^2+\frac{4\hat{\mu}}{\xi}}}{2} = & \frac{\frac{2\hat{\mu}}{\xi}}{x_i^*+s_i^*+\sqrt{(x_i^*+s_i^*)^2+\frac{4\hat{\mu}}{\xi}}} = (\hat{\lambda}_U)_i, \end{cases} \tag{4.12}$$

for all $i \in \{1, \ldots, n\}$. For any $\xi \in (0,1)$, it is easy to see that (4.12) yields a well-defined interval for $\lambda_i$, $i \in \{1, \ldots, n\}$. □

Note that (4.10) is the symmetric neighbourhood (2.11) for the perturbed problems.

From the above theorem, we see that by choosing the perturbations judiciously, we can bring any solution of the original problem into a 'neighbourhood' of the perturbed central path. This implies that we can generate a series of iterates along the central path of the perturbed problems and may have a point on the central path of the perturbed problems that is very close to the original optimal solution but still far from the optimal boundaries of the perturbed problems.

### 4.2.2 Preserving the optimal active set

Since we are interested in predicting the optimal active set of the original problem, this section addresses the relation between the active set of the perturbed problem and that of the original LP. We find that for sufficiently small perturbations, these two active sets remain the same provided the original problem is nondegenerate.

**Theorem 4.7 (Preserving the optimal active set).** *Assume* (2.9) *holds and the original pair of* (PD) *problems has a unique and nondegenerate primal solution*

*$x^*$. Then there exists a positive scalar $\hat{\lambda} = \hat{\lambda}(A, b, c, x^*)$ such that the pair of perturbed problems ($PD_\lambda$) with $0 \leq \|\lambda\| < \hat{\lambda}$ has a strictly complementary solution $(x^*_\lambda, y^*_\lambda, s^*_\lambda)$ with the same active and inactive sets as $x^*$.*[10]

**Proof.** Since (PD) has a unique and nondegenerate primal solution, it must have a unique primal-dual nondegenerate solution $(x^*, y^*, s^*)$ (Proposition b in Theorem 2.5), which must be strictly complementary (due to Theorem 2.3) and so $x^* + s^* > 0$. Thus, letting $(\mathcal{A}, \mathcal{I})$ be the strictly complementary partition defined in (2.7), the KKT conditions (2.3) for (PD) at $(x^*, y^*, s^*)$ become

$$x^*_{\mathcal{A}} = 0, \quad x^*_{\mathcal{I}} > 0 \quad \text{and} \quad s^*_{\mathcal{I}} = 0, \quad s^*_{\mathcal{A}} > 0, \tag{4.13a}$$

$$A_{\mathcal{I}} x^*_{\mathcal{I}} = b, \quad A^T_{\mathcal{I}} y^* = c_{\mathcal{I}}, \quad A^T_{\mathcal{A}} y^* + s^*_{\mathcal{A}} = c_{\mathcal{A}}, \tag{4.13b}$$

where $A = [A_{\mathcal{I}}\ A_{\mathcal{A}}]$, $x^* = (x^*_{\mathcal{A}}, x^*_{\mathcal{I}})$ and $s^* = (s^*_{\mathcal{A}}, s^*_{\mathcal{I}})$. As the (PD) solution is also nondegenerate, we must have $|\mathcal{I}| = m$ and $rank(A_{\mathcal{I}}) = m$, namely, $A_{\mathcal{I}}$ is nonsingular. We work with the equivalent form (4.6) of problems ($PD_\lambda$), and construct a solution $(\hat{p}, \hat{y}, \hat{q})$ of (4.6) such that $\hat{p} + \hat{q} > 0$, $\hat{p}_{\mathcal{A}} = 0$ and $\hat{q}_{\mathcal{I}} = 0$, namely,

$$\hat{p}_{\mathcal{A}} = 0, \quad \hat{p}_{\mathcal{I}} = x^*_{\mathcal{I}} + \lambda_{\mathcal{I}} + A^{-1}_{\mathcal{I}} A_{\mathcal{A}} \lambda_{\mathcal{A}}, \tag{4.14a}$$

$$\hat{y} = y^* + (A^T_{\mathcal{I}})^{-1} \lambda_{\mathcal{I}}, \quad \hat{q}_{\mathcal{I}} = 0, \quad \hat{q}_{\mathcal{A}} = s^*_{\mathcal{A}} + \lambda_{\mathcal{A}} - (A^{-1}_{\mathcal{I}} A_{\mathcal{A}})^T \lambda_{\mathcal{I}}. \tag{4.14b}$$

Using (4.13), it is straightforward to show that $(\hat{p}, \hat{y}, \hat{q})$ in (4.14) satisfies all linear and nonlinear equality constraints in the KKT conditions (4.7). It remains to prove that $\hat{p}_{\mathcal{I}} > 0$ and $\hat{q}_{\mathcal{A}} > 0$. Let $\sigma_{\max}$ be the largest singular value of $A^{-1}_{\mathcal{I}} A_{\mathcal{A}}$, and define a positive scalar $\hat{\lambda}$ as

$$\hat{\lambda} = \frac{\min [x^*_{\mathcal{I}}\ s^*_{\mathcal{A}}]}{\sigma_{\max}}, \tag{4.15}$$

where $\min [x^*_{\mathcal{I}}\ s^*_{\mathcal{A}}]$ is a scalar that denotes the smallest element of $x^*_{\mathcal{I}}$ and $s^*_{\mathcal{A}}$. From $\lambda \geq 0$ and from norm properties, we have that

$$\begin{aligned} \hat{p}_{\mathcal{I}} \geq x^*_{\mathcal{I}} - \|A^{-1}_{\mathcal{I}} A_{\mathcal{A}} \lambda_{\mathcal{A}}\| e_{\mathcal{I}} &\geq x^*_{\mathcal{I}} - \|A^{-1}_{\mathcal{I}} A_{\mathcal{A}}\| \cdot \|\lambda_{\mathcal{A}}\| e_{\mathcal{I}} \\ &\geq x^*_{\mathcal{I}} - \|A^{-1}_{\mathcal{I}} A_{\mathcal{A}}\| \cdot \|\lambda\| e_{\mathcal{I}} \end{aligned}$$

and

$$\begin{aligned} \hat{q}_{\mathcal{A}} \geq s^*_{\mathcal{A}} - \| \left(A^{-1}_{\mathcal{I}} A_{\mathcal{A}}\right)^T \lambda_{\mathcal{I}}\| e_{\mathcal{A}} &\geq s^*_{\mathcal{A}} - \| \left(A^{-1}_{\mathcal{I}} A_{\mathcal{A}}\right)^T \| \cdot \|\lambda_{\mathcal{I}}\| e_{\mathcal{A}} \\ &\geq s^*_{\mathcal{A}} - \| \left(A^{-1}_{\mathcal{I}} A_{\mathcal{A}}\right)^T \| \cdot \|\lambda\| e_{\mathcal{A}}. \end{aligned}$$

Using matrix norm properties, we obtain that $\left\|A^{-1}_{\mathcal{I}} A_{\mathcal{A}}\right\| = \left\|(A^{-1}_{\mathcal{I}} A_{\mathcal{A}})^T\right\| = \sigma_{\max}$. This and $0 < \|\lambda\| < \hat{\lambda}$ now imply

$$\hat{p}_{\mathcal{I}} > x^*_{\mathcal{I}} - \sigma_{\max} \hat{\lambda} e_{\mathcal{I}} \geq x^*_{\mathcal{I}} - \min [x^*_{\mathcal{I}}\ s^*_{\mathcal{A}}]\, e_{\mathcal{I}} \geq 0,$$

[10] The norm $\|\cdot\|$ denotes the Euclidean norm.

and

$$\hat{q}_{\mathcal{A}} > s^*_{\mathcal{A}} - \sigma_{\max}\hat{\lambda}e_{\mathcal{A}} \geq s^*_{\mathcal{A}} - \min\left[x^*_{\mathcal{I}}\ s^*_{\mathcal{A}}\right] e_{\mathcal{A}} \geq 0,$$

where we also use the definition of $\hat{\lambda}$. □

An equivalent non-degeneracy assumption that would be sufficient here is to require that all (PD) solutions are primal-dual nondegenerate (see [63, Section 5] or Theorem 2.5).

**Remark on the assumptions and proof of Theorem 4.7.** We have assumed in this theorem that (PD) is primal-dual nondegenerate and has a unique solution, which guarantees $A_{\mathcal{I}}$ is nonsingular. Considering the general case when $(x^*, y^*, s^*)$ is a possibly non-unique strictly complementary solution, to construct the desired solution $(\hat{p}, \hat{y}, \hat{q})$ of (4.6) with the same active set and strictly complementary partition, one needs to satisfy exactly primal-dual feasibility requirements such as

$$A_{\mathcal{I}}\hat{p}_{\mathcal{I}} = b + A\lambda = b + A_{\mathcal{A}}\lambda_{\mathcal{A}} + A_{\mathcal{I}}\lambda_{\mathcal{I}}. \tag{4.16}$$

Clearly, one can only guarantee (4.16) to be consistent for $\lambda > 0$ if $A_{\mathcal{A}}\lambda_{\mathcal{A}}$ belongs to the range space of $A_{\mathcal{I}}$. Alternatively, one could consider satisfying (4.16) only approximately and look for a solution $\hat{p}$ of the form

$$\hat{p}_{\mathcal{A}} = 0 \quad \text{and} \quad \hat{p}_{\mathcal{I}} = x^*_{\mathcal{I}} + \lambda_{\mathcal{I}} + \hat{u}, \tag{4.17}$$

where $\hat{u}$ is the least-squares/minimal norm solution of $A_{\mathcal{I}}u = A_{\mathcal{A}}\lambda_{\mathcal{A}}$. For instance in the case when $|\mathcal{I}| \leq m$, we have $\|A_{\mathcal{I}}\hat{u} - A_{\mathcal{A}}\lambda_{\mathcal{A}}\| \leq \|A_{\mathcal{A}}\lambda_{\mathcal{A}}\|$. The right-hand side of the latter inequality goes to zero as $\lambda \to 0$ and so primal feasibility can be approximately achieved. It can also be shown that $\hat{p}_{\mathcal{I}}$ in (4.17) stays positive. We will employ this idea for QP problems and prove a general result in Section 7.2.2; see Theorem 7.4 and its proof for details. □

Note that the nondegeneracy assumption in Theorem 4.7 is not required in the results of the next chapter or in our implementations and numerical experiments. Thus this theorem and its assumptions do not restrict our algorithmic or even main theoretical approach of predicting the optimal active set of the (PD) problem by solving a perturbed ($\text{PD}_\lambda$) problem.

**A simple example of preserving the optimal active set.** To illustrate Theorem 4.7, we consider the following example,

$$\begin{array}{lrrrrrl} \min & -5x_1 & -4x_2, & & & & \\ \text{s.t.} & 2x_1 & +x_2 & +x_3 & & & = 8, \\ & -3x_1 & +2x_2 & & +x_4 & & = 6, \\ & x_1 & +2x_2 & & & +x_5 & = 6, \\ & x_i \geq 0, & i = 1, \ldots, 5. & & & & \end{array} \tag{4.18}$$

The unique solution of (4.18) is $x^* = (\frac{10}{3}, \frac{4}{3}, 0, \frac{40}{3}, 0)$ and the corresponding dual solution is $y^* = (2, 0, 1)$ and $s^* = (0, 0, 2, 0, 1)$. So $\mathcal{A} = \{3, 5\}$ and $\mathcal{I} = \{1, 2, 4\}$,

$$A_{\mathcal{I}} = \begin{bmatrix} 2 & 1 & 0 \\ -3 & 2 & 1 \\ 1 & 2 & 0 \end{bmatrix} \quad \text{and} \quad A_{\mathcal{A}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

And thus $\sigma_{\max} \approx 3.6813$ (the largest singular value of $(A_{\mathcal{I}})^{-1}A_{\mathcal{A}}$). From (4.15), we have that the largest value of the perturbations is

$$\hat{\lambda} \approx \frac{\min(\frac{10}{3}, \frac{4}{3}, \frac{40}{3}, 2, 1)}{3.6813} \approx 0.2716.$$

If we choose $\lambda_i = 0.1$ for all $i = 1, \ldots, 5$, all conditions in Theorem 4.7 are satisfied. Solving the corresponding perturbed primal problem

$$\begin{array}{lllllll} \min & (-5+\lambda_1)x_1 & +(-4+\lambda_2)x_2 & +\lambda_3 x_3 & +\lambda_4 x_4 & +\lambda_5 x_5, & \\ \text{s.t.} & 2x_1 & +x_2 & +x_3 & & & = 8, \\ & -3x_1 & +2x_2 & & +x_4 & & = 6, \\ & x_1 & +2x_2 & & & +x_5 & = 6, \\ & x_i \geq -\lambda_i,\ i = 1, \ldots, 5, & & & & & \end{array}$$

and its dual, we deduce that the primal-dual optimal solution of the perturbed problems is

$$x^*_\lambda = (\frac{10.1}{3}, \frac{4.1}{3}, -0.1, \frac{40.1}{3}, -0.1), \quad y^*_\lambda = (1.7, -0.1, 1.2), \quad s^*_\lambda = (-0.1, -0.1, 1.7, -0.1, 1.2).$$

Clearly, the strictly complementary partition of ($\text{PD}_\lambda$), $(\mathcal{A}_\lambda, \mathcal{I}_\lambda)$ in (4.5) coincides with $(\mathcal{A}, \mathcal{I})$ in (2.7), the strictly complementary partition of (PD). □

This example is only meant as an illustration of the theory, not the implementation or numerical results, for which we do not require the assumption of a unique solution or that the optimal perturbed and unperturbed active sets coincide.

# 5 Active-set Prediction Using Controlled Perturbations

In this chapter, recalling our main aim, we present results for predicting the optimal active set of (PD). We solve a perturbed problem of the form ($PD_\lambda$) instead of the original one using IPMs, but attempt to predict the active set for the original problem during the run of the algorithm. Some useful results are derived in Section 5.1. We prove that under certain conditions and given proper perturbations, when the duality gap of ($PD_\lambda$) is sufficiently small, the predicted (strongly) active set for (PD) using cut-off coincides with the actual optimal (strongly) active set of (PD) (Theorems 5.7, 5.8 in Section 5.2). We also find conditions on problem conditioning that ensure that our prediction based on cut-off of the optimal active set of the original LP can happen sooner than the prediction of the optimal active set of the perturbed problems (so that our approach may not need to solve the perturbed problems to high accuracy) (Theorem 5.10 in Section 5.3). Similarly, we characterise the situations when our approach allows an earlier prediction of the original active set as compared to the case when we solve and predict the original LP directly (Theorem 5.13 in Section 5.3). At the end of this chapter (Section 5.5), we derive a similar set of prediction results but using the identification function (defined in Section 3.1) instead of cut-off for active-set prediction.

## 5.1 Some useful results

We first derive a bound on the distance between the original optimal solution set and strictly feasible points of the perturbed problems ($PD_\lambda$).

**Lemma 5.1.** *Let $(x, y, s) \in \mathcal{F}_\lambda^0$, where $\mathcal{F}_\lambda^0$ is defined in (4.1), and $\lambda \geq 0$. Then there exists a (PD) solution $(x^*, y^*, s^*)$ such that*

$$\|x - x^*\| \leq \tau_p \left(r(x, s) + w(x, s)\right) \quad \text{and} \quad \|s - s^*\| \leq \tau_d \left(r(x, s) + w(x, s)\right), \qquad \text{(ER)}$$

*where $\tau_p > 0$ and $\tau_d > 0$ are problem-dependent constants independent of $(x, y, s)$ and*

$(x^*, y^*, s^*)$, *and*

$$r(x,s) = \| \min\{x,s\} \| \quad \textit{and} \quad w(x,s) = \|(-x,-s,x^T s)_+\|, \tag{5.1}$$

*and where* $\min\{x,s\} = (\min(x_i,s_i))_{i=1,\ldots,n}$ *and* $(x)_+ = (\max(x_i,0))_{i=1,\ldots,n}$.

**Proof.** Since $(x,y,s) \in \mathcal{F}_\lambda^0$, (4.1) gives $Ax = b$ and $A^T y + s = c$. Then the result follows directly from Theorem 2.16. □

Recall the quantity $\epsilon(A,b,c)$ in (2.22) for (PD), which has been defined in Lemma 2.8 and used to construct an upper bound for some components of $x$ and $s$ (see page 24). Similarly, for ($\text{PD}_\lambda$) we denote

$$\epsilon(A,b_\lambda,c_\lambda) = \min\left( \min_{i\in\mathcal{I}_\lambda} \sup_{x_\lambda^*\in\Omega_\lambda^P} \{ (x_\lambda^*)_i + \lambda_i \}, \min_{i\in\mathcal{A}_\lambda} \sup_{(y_\lambda^*,s_\lambda^*)\in\Omega_\lambda^D} \{ (s_\lambda^*)_i + \lambda_i \} \right) > 0, \tag{5.2}$$

where $\Omega_\lambda^P$ and $\Omega_\lambda^D$ are the primal and dual solution sets of ($\text{PD}_\lambda$) defined in (4.4) and $(\mathcal{A}_\lambda, \mathcal{I}_\lambda)$ is the strictly complementary partition for ($\text{PD}_\lambda$) defined in (4.5). When the feasible set of (PD) is nonempty, that of ($\text{PD}_\lambda$) is also nonempty, and so, from the proof of Lemma 2.8, $\epsilon(A,b_\lambda,c_\lambda) > 0$. Next we rephrase Lemma 5.13 in [135], apply it to ($\text{PD}_\lambda$), and show that $\epsilon(A,b_\lambda,c_\lambda)$ can also be used to construct an upper bound for some components of $x+\lambda$ and $s+\lambda$; note that [135, Lemma 5.13] has been reproduced in this thesis for (PD) as Lemmas 2.8 and 2.9.

**Lemma 5.2.** *For any* $(x,y,s) \in \mathcal{F}_\lambda^0$, *where* $\mathcal{F}_\lambda^0$ *is defined in* (4.1), *we have*

$$0 < x_i + \lambda_i \le \frac{\mu_\lambda}{C_1} \quad (i \in \mathcal{A}_\lambda) \quad \textit{and} \quad 0 < s_i + \lambda_i \le \frac{\mu_\lambda}{C_1} \quad (i \in \mathcal{I}_\lambda), \tag{5.3}$$

*where*

$$\mu_\lambda = \frac{(x+\lambda)^T(s+\lambda)}{n} \tag{5.4}$$

*and*

$$C_1 = \frac{\epsilon(A,b_\lambda,c_\lambda)}{n} \tag{5.5}$$

*with* $\epsilon(A,b_\lambda,c_\lambda)$ *defined in* (5.2) *and* $(\mathcal{A}_\lambda,\mathcal{I}_\lambda)$ *in* (4.5).

**Proof.** Apply Lemma 2.8 to (4.6) and recall $x = p - \lambda$ and $s = q - \lambda$. □

In Lemmas 5.3 and 5.4, we derive upper bounds for $r(x,s)$ and $w(x,s)$, in order to further derive upper bounds for $\|x - x^*\|$ and $\|s - s^*\|$ in Lemma 5.5.

**Lemma 5.3.** *Let* $(x,y,s) \in \mathcal{F}_\lambda^0$, *where* $\mathcal{F}_\lambda^0$ *is defined in* (4.1) *for some* $\lambda \ge 0$. *Then*

$$r(x,s) \le \frac{\mu_\lambda}{C_1}\sqrt{n} + \|\lambda\|, \tag{5.6}$$

where $r(x,s)$ is defined in (5.1) and $\mu_\lambda$ in (5.4).

**Proof.** If $i \in \mathcal{A}_\lambda$, where $\mathcal{A}_\lambda$ is defined in (4.5), from (5.3) we have

$$\min\left(x_i + \lambda_i, s_i + \lambda_i\right) \leq x_i + \lambda_i \leq \frac{\mu_\lambda}{C_1}.$$

Similarly, we also have $\min\left(x_i + \lambda_i, s_i + \lambda_i\right) \leq \frac{\mu_\lambda}{C_1}$ for $i \in \mathcal{I}_\lambda$. Thus

$$0 < \min\left\{x + \lambda, s + \lambda\right\} \leq \frac{\mu_\lambda}{C_1} e,$$

and so from (5.1),

$$r(x,s) = \|\min\left\{x + \lambda, s + \lambda\right\} - \lambda\| \leq \|\min\left\{x + \lambda, s + \lambda\right\}\| + \|\lambda\| \leq \frac{\mu_\lambda}{C_1}\sqrt{n} + \|\lambda\|.$$

□

**Lemma 5.4.** *Let $(x, y, s) \in \mathcal{F}_\lambda^0$, where $\mathcal{F}_\lambda^0$ is defined in (4.1) for some $\lambda \geq 0$. Then*

$$w(x,s) \leq n\mu_\lambda + 2\|\lambda\| + \|\lambda\|^2, \tag{5.7}$$

*where $w(x,s)$ is defined in (5.1) and $\mu_\lambda$ in (5.4).*

**Proof.** Since $x + \lambda > 0$ and $s + \lambda > 0$, we have $-x < \lambda$ and $-s < \lambda$, which implies

$$0 \leq (-x)_+ < \lambda \quad \text{and} \quad 0 \leq (-s)_+ < \lambda. \tag{5.8}$$

Using (5.4), $\lambda \geq 0$ and $(x + \lambda, s + \lambda) \geq 0$, we have

$$x^T s = n\mu_\lambda + \lambda^T\lambda - \lambda^T(x + \lambda) - \lambda^T(s + \lambda) \leq n\mu_\lambda + \|\lambda\|^2. \tag{5.9}$$

From (5.1), (5.8) and (5.9), we obtain

$$w(x,s) \leq \|(-x)_+\| + \|(-s)_+\| + (x^T s)_+ \leq n\mu_\lambda + 2\|\lambda\| + \|\lambda\|^2.$$

□

**Lemma 5.5.** *Let $(x, y, s) \in \mathcal{F}_\lambda^0$, where $\mathcal{F}_\lambda^0$ is defined in (4.1) for some $\lambda \geq 0$. Then there exists a (PD) solution $(x^*, y^*, s^*)$ and problem-dependent constants $\tau_p$ and $\tau_d$ that are independent of $(x, y, s)$ and $(x^*, y^*, s^*)$, such that*

$$\|x - x^*\| < \tau_p\left(C_2\mu_\lambda + 4\|\lambda\|\max\left(\|\lambda\|, 1\right)\right) \text{ and } \|s - s^*\| < \tau_d\left(C_2\mu_\lambda + 4\|\lambda\|\max\left(\|\lambda\|, 1\right)\right), \tag{5.10}$$

*where*

$$C_2 = \frac{n\sqrt{n}}{\epsilon(A, b_\lambda, c_\lambda)} + n, \tag{5.11}$$

*$\epsilon(A, b_\lambda, c_\lambda)$ is defined in (5.2) and $\mu_\lambda$ in (5.4).*

**Proof.** The bound (5.10) follows from (ER) in Lemma 5.1, (5.6) and (5.7). □

## 5.2 Predicting the original optimal active set using perturbations

During the iterative process of solving the perturbed problem using an interior point framework, we try to predict the optimal active set for the original problem. Assume $(x^*, y^*, s^*)$ is a (PD) solution. We recall that $\mathcal{A}(x^*)$, defined in (2.6a), denotes the optimal active set at $(x^*, y^*, s^*)$ and $\mathcal{A}^+(s^*)$ in (2.6b) stands for the strongly active set. Let

$$\bar{\mathcal{A}}(x) = \{i \in \{1, \dots, n\} \,|\, x_i < C\} \quad \text{and} \quad \bar{\mathcal{A}}^+(s) = \{i \in \{1, \dots, n\} \,|\, s_i \geq C\}, \tag{5.12}$$

where $C > 0$ is some constant threshold. $\bar{\mathcal{A}}(x)$ is considered as the predicted active set and $\bar{\mathcal{A}}^+(s)$, the predicted strongly active set at a primal-dual pair $(x, y, s)$ for ($\text{PD}_\lambda$).

**Theorem 5.6.** *Let $C > 0$ and fix the perturbation $\lambda$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{C}{8\max(\tau_p, \tau_d)}\right), \tag{5.13}$$

*where $\tau_p$ and $\tau_d$ are the problem-dependent constants in (5.10). Let $(x, y, s) \in \mathcal{F}_\lambda^0$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \frac{C}{2C_2\max(\tau_p, \tau_d)}, \tag{5.14}$$

*where $\mathcal{F}_\lambda^0$ is defined in (4.1), $\mu_\lambda$ in (5.4) and $C_2 > 0$ in (5.11) is a problem-dependent constant when $\lambda$ is fixed. Then there exists a (PD) solution $(x^*, y^*, s^*)$ such that*

$$\bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*) \subseteq \bar{\mathcal{A}}(x).$$

**Proof.** From $\|\lambda\| < 1$ and (5.10), we have

$$\|x - x^*\| \leq \tau_p\left(C_2\mu_\lambda + 4\|\lambda\|\right) \quad \text{and} \quad \|s - s^*\| \leq \tau_d\left(C_2\mu_\lambda + 4\|\lambda\|\right),$$

which imply

$$x_i^* - \tau_p\left(C_2\mu_\lambda + 4\|\lambda\|\right) \leq x_i \leq x_i^* + \tau_p\left(C_2\mu_\lambda + 4\|\lambda\|\right) \tag{5.15}$$

and

$$s_i^* - \tau_d\left(C_2\mu_\lambda + 4\|\lambda\|\right) \leq s_i \leq s_i^* + \tau_d\left(C_2\mu_\lambda + 4\|\lambda\|\right), \tag{5.16}$$

for all $i \in \{1, \ldots, n\}$. If $i \in \mathcal{A}(x^*)$, from (5.13), (5.14) and (5.15), we have $x_i < C$, namely $i \in \bar{\mathcal{A}}(x)$. So $\mathcal{A}(x^*) \subseteq \bar{\mathcal{A}}(x)$. If $i \notin \mathcal{A}^+(s^*)$, $s_i^* = 0$. Then from (5.13), (5.14) and (5.16), we have $s_i < C$, namely, $i \notin \bar{\mathcal{A}}^+(s)$. Thus $\bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}^+(s^*)$. From $x_i^* s_i^* = 0$ for all $i \in \{1, \ldots, n\}$, we have $\mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*)$. □

Theorem 5.6 shows that $\bar{\mathcal{A}}(x)$ and $\bar{\mathcal{A}}^+(s)$ serve as a pair of approximations that bound $\mathcal{A}(x^*)$. Next we go a step further and show that $\bar{\mathcal{A}}(x)$ is equivalent to $\mathcal{A}(x^*)$ under certain conditions.

**Theorem 5.7.** *Let*

$$\psi_p = \inf_{x^* \in \Omega^P} \min_{i \notin \mathcal{A}(x^*)} (x_i^*) \tag{5.17}$$

*where $\Omega^P$ is the solution set of the primal problem in (PD) defined in (2.4a) and $\mathcal{A}(x^*)$ is defined in (2.6a). Assume $\psi_p > 0$. Fix $\lambda$ and $C$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{\psi_p}{16 \max(\tau_p, \tau_d)}\right) \quad \text{and} \quad C = \frac{\psi_p}{2}, \tag{5.18}$$

*where $\tau_p$ and $\tau_d$ are the problem-dependent constants defined in (5.10). Let $(x, y, s) \in \mathcal{F}_\lambda^0$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \frac{\psi_p}{4C_2 \max(\tau_p, \tau_d)}, \tag{5.19}$$

*where $\mathcal{F}_\lambda^0$ is defined in (4.1), $\mu_\lambda$ in (5.4) and $C_2 > 0$ in (5.11). Then there exists a (PD) solution $(x^*, y^*, s^*)$ such that*

$$\bar{\mathcal{A}}(x) = \mathcal{A}(x^*),$$

*where $\bar{\mathcal{A}}(x)$ is defined in (5.12).*

**Proof.** From Theorem 5.6 we have $\mathcal{A}(x^*) \subseteq \bar{\mathcal{A}}(x)$. It remains to prove $\bar{\mathcal{A}}(x) \subseteq \mathcal{A}(x^*)$. If $i \notin \mathcal{A}(x^*)$, from the left inequality in (5.15), (5.18) and (5.19), we have

$$x_i > x_i^* - \frac{\psi_p}{2} \cdot \frac{\tau_p}{\max(\tau_p, \tau_d)} \geq \inf_{x^* \in \Omega^P} \min_{i \notin \mathcal{A}(x^*)} (x_i^*) - \frac{\psi_p}{2} = \psi_p - \frac{\psi_p}{2} = C.$$

Thus $i \notin \bar{\mathcal{A}}(x)$, which implies $\bar{\mathcal{A}}(x) \subseteq \mathcal{A}(x^*)$. □

Next, we show that $\bar{\mathcal{A}}^+(s)$, the predicted strongly active set at a strictly feasible point $(x, y, s)$ of ($\text{PD}_\lambda$), is the same as $\mathcal{A}^+(s^*)$ at some (PD) solution $(x^*, y^*, s^*)$.

**Theorem 5.8.** *Let*

$$\psi_d = \inf_{(y^*,s^*)\in\Omega^D} \min_{i\in\mathcal{A}^+(s^*)} (s_i^*), \tag{5.20}$$

*where* $\Omega^D$ *is the solution set of the dual problem in* (PD) *defined in* (2.4b) *and* $\mathcal{A}^+(s^*)$ *is defined in* (2.6b). *Assume* $\psi_d > 0$. *Fix* $\lambda$ *and* $C$ *such that*

$$0 < \|\lambda\| < \min\left(1, \frac{\psi_d}{16\max(\tau_p,\tau_d)}\right) \quad \text{and} \quad C = \frac{\psi_d}{2}, \tag{5.21}$$

*where* $\tau_p$ *and* $\tau_d$ *are the problem-dependent constants in* (5.10). *Let* $(x,y,s) \in \mathcal{F}_\lambda^0$ *with* $\mu_\lambda$ *sufficiently small, namely*

$$\mu_\lambda < \frac{\psi_d}{4C_2\max(\tau_p,\tau_d)}, \tag{5.22}$$

*where* $\mathcal{F}_\lambda^0$ *is defined in* (4.1), $\mu_\lambda$ *in* (5.4) *and* $C_2 > 0$ *in* (5.11). *Then there exists a* (PD) *solution* $(x^*,y^*,s^*)$ *such that*

$$\bar{\mathcal{A}}^+(s) = \mathcal{A}^+(s^*),$$

*where* $\bar{\mathcal{A}}^+(s)$ *is defined in* (5.12).

**Proof.** From Theorem 5.6, we have $\bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}^+(s^*)$. If $i \in \mathcal{A}^+(s^*)$, $s_i^* > 0$. This, (5.16), (5.21) and (5.22) give us

$$s_i > s_i^* - \frac{\psi_d}{2}\cdot\frac{\tau_d}{\max(\tau_p,\tau_d)} \geq \inf_{(y^*,s^*)\in\Omega^D} \min_{i\in\mathcal{A}^+(s^*)} (s_i^*) - \frac{\psi_d}{2} = \psi_d - \frac{\psi_d}{2} = C,$$

namely $\mathcal{A}^+(s^*) \subseteq \bar{\mathcal{A}}^+(s)$. □

**Remarks on Theorems 5.6–5.8.**

- We require $\mu_\lambda$, the mean value of the complementary products, to be sufficiently small in Theorems 5.6–5.8. This choice is possible since we have $\mu_\lambda = 0$ at any optimal solution of ($\text{PD}_\lambda$) and $\mu_\lambda$ can be decreased to zero (such as in an IPM framework).
- Theorems 5.7 and 5.8 state that any feasible point for ($\text{PD}_\lambda$) for small enough $\mu_\lambda$ and $\lambda$ cannot be far away from the original optimal solution $(x^*,y^*,s^*)$ and thus have the same active set as $(x^*,y^*,s^*)$; since such a point can also be the optimal solution to ($\text{PD}_\lambda$), it likewise implies that (PD) and ($\text{PD}_\lambda$) have the same optimal active set.
- $\psi_p$ in (5.17) is positive if the primal problem in (PD) has a unique (degenerate or nondegenerate) solution, but we expect that it may often be zero in the case of

multiple solutions. (Clearly, in our implementations, we do not choose the cut-off value based on the theoretical quantity $\psi_p$.) Similarly to $\psi_p$, if the dual problem in (PD) has a unique (degenerate or nondegenerate) solution, we have $\psi_d > 0$.

- Fix $\lambda$ sufficiently small and let $(x^k, y^k, s^k)$ be iterates of a primal-dual path-following IPM applied to (PD$_\lambda$). Then assuming these iterates belong to some good neighbourhood of the central path of (PD$_\lambda$) and that the barrier parameter is decreased appropriately, we have $\mu_\lambda^k \to 0$ as $k \to \infty$ [135, Theorem 5.11]. So, by applying Theorem 5.7, for each $k$ sufficiently large, there exists a (PD) solution $(x^*, y^*, s^*)$ such that $\bar{\mathcal{A}}(x^k) = \mathcal{A}(x^*)$ (see also Lemma 5.9 below).

## 5.3 Comparing perturbed and unperturbed active-set predictions

### 5.3.1 Comparing with active-set prediction for (PD$_\lambda$)

Consider the 'large' neighbourhood of the perturbed central path

$$\mathcal{N}_{-\infty}(\gamma, \lambda) = \{\, (x, y, s) \in \mathcal{F}_\lambda^0 \mid (x_i + \lambda_i)(s_i + \lambda_i) \geq \gamma \mu_\lambda,\ i = 1, \ldots, n \,\}, \tag{5.23}$$

where $\mathcal{F}_\lambda^0$ is defined in (4.1) and $\mu_\lambda$ is defined in (5.4); see (2.10) for the definition (5.23) in the case of $\lambda \equiv 0$ and also the discussions on different neighbourhoods for IPMs in the same section.

Next we rephrase Lemma 5.13 in [135] as an active-set prediction result for (PD$_\lambda$); this lemma has been reproduced here for (PD) as Lemmas 2.8 and 2.9.

**Lemma 5.9.** *Let $(x, y, s)$ in $\mathcal{N}_{-\infty}(\gamma, \lambda)$ and $\mu_\lambda$ defined in (5.4). Assume $C$ in (5.12) is set to $C = \frac{\epsilon(A, b_\lambda, c_\lambda)\gamma}{n}$, where $\epsilon(A, b_\lambda, c_\lambda)$ is defined in (5.2). Then when $\mu_\lambda < \bar{\mu}_\lambda^{\max}$, where*

$$\bar{\mu}_\lambda^{\max} = \frac{\epsilon^2(A, b_\lambda, c_\lambda)\gamma}{n^2}, \tag{5.24}$$

*for any strictly complementary solution $(x_\lambda^*, y_\lambda^*, s_\lambda^*)$ of (PD$_\lambda$) we have*

$$\bar{\mathcal{A}}(x + \lambda) = \mathcal{A}(x_\lambda^* + \lambda),$$

*where $\bar{\mathcal{A}}(x + \lambda)$ is defined in (5.12) with $x$ replaced by $x + \lambda$ and $\mathcal{A}(x_\lambda^* + \lambda)$ is defined in (2.6) with $x^*$ replaced by $x_\lambda^* + \lambda$.*

**Proof.** We work with the equivalent form (4.6) of (PD$_\lambda$). Given (5.24), apply [135, Lemma 5.13] to (4.6), or equivalently Lemmas 2.8 and 2.9, recalling that $x = p - \lambda$ and $s = q - \lambda$, and then we have

$$\begin{aligned} i \in \mathcal{A}_\lambda: &\quad 0 < x_i + \lambda_i \leq \tfrac{\mu_\lambda}{C_1} < C_1\gamma \leq s_i + \lambda_i, \\ i \in \mathcal{I}_\lambda: &\quad 0 < s_i + \lambda_i \leq \tfrac{\mu_\lambda}{C_1} < C_1\gamma \leq x_i + \lambda_i, \end{aligned} \tag{5.25}$$

where $\mathcal{A}_\lambda$ and $\mathcal{I}_\lambda$ are defined in (4.5). For any strictly complementary solution $(x^*_\lambda, y^*_\lambda, s^*_\lambda)$ of ($\text{PD}_\lambda$), $(x^*_\lambda+\lambda, y^*_\lambda, s^*_\lambda+\lambda)$ is a strictly complementary solution of (4.6). This and the definition of $\mathcal{A}(x^*_\lambda+\lambda)$ give us that $\mathcal{A}(x^*_\lambda+\lambda) = \mathcal{A}_\lambda$. From (5.25) and the definition of $\bar{\mathcal{A}}(x+\lambda)$, we also have $\bar{\mathcal{A}}(x+\lambda) = \mathcal{A}_\lambda$. □

Substituting (5.11) into (5.19), we obtain the following threshold value

$$\mu_\lambda^{\max} := \frac{\psi_p \epsilon(A, b_\lambda, c_\lambda)}{4n \max(\tau_p, \tau_d)\left(\sqrt{n} + \epsilon(A, b_\lambda, c_\lambda)\right)}, \tag{5.26}$$

where $\epsilon(A, b_\lambda, c_\lambda)$ is defined in (5.2), $\psi_p$ in (5.17), and $\tau_p$ and $\tau_d$ are the positive constants in the bounds (5.10). Theorem 5.7 provides that when $\psi_p > 0$ and $\lambda$ is sufficiently small and fixed, if $\mu_\lambda < \mu_\lambda^{\max}$, we can predict the optimal active set of (PD). Lemma 5.9 shows that when $\mu_\lambda < \bar{\mu}_\lambda^{\max}$, where $\bar{\mu}_\lambda^{\max}$ is defined in (5.24), we can provide the strictly complementary partition of the solution set of ($\text{PD}_\lambda$) from any primal-dual pair in the neighbourhood $\mathcal{N}_{-\infty}(\gamma, \lambda)$ of the perturbed central path. To verify if our approach can predict the optimal active set of (PD) before the strictly complementary partition of ($\text{PD}_\lambda$), we determine conditions under which $\mu_\lambda^{\max} > \bar{\mu}_\lambda^{\max}$.

**Theorem 5.10.** *In the conditions of Theorem 5.7, let*

$$\kappa = \frac{\psi_p}{\max(\tau_p, \tau_d)}. \tag{5.27}$$

*If*

$$\epsilon(A, b_\lambda, c_\lambda) \leq \mathcal{O}\left(\sqrt{n\kappa}\min\left(\sqrt{\kappa}, 1\right)\right), \tag{5.28}$$

*then*

$$\mu_\lambda^{\max} > \bar{\mu}_\lambda^{\max},$$

*where $\epsilon(A, b_\lambda, c_\lambda)$ is defined in (5.2), $\mu_\lambda^{\max}$ in (5.26) and $\bar{\mu}_\lambda^{\max}$ in (5.24).*

**Proof.** Note that $\mu_\lambda^{\max} > \bar{\mu}_\lambda^{\max}$ is equivalent to

$$\epsilon^2(A, b_\lambda, c_\lambda) + \sqrt{n}\epsilon(A, b_\lambda, c_\lambda) - \frac{\kappa}{4\gamma} < 0,$$

which is satisfied if

$$0 < \epsilon(A, b_\lambda, c_\lambda) \leq \frac{\sqrt{n}}{2\sqrt{\gamma}} \cdot \frac{\kappa}{\sqrt{\gamma+\kappa} + \sqrt{\gamma}}. \tag{5.29}$$

Since $\gamma \in (0, 1)$ and $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for any $a$ and $b$ nonnegative scalars, we have

$$\frac{\kappa}{\sqrt{\gamma+\kappa} + \sqrt{\gamma}} \geq \frac{\kappa}{\sqrt{\kappa} + 2\sqrt{\gamma}} \geq \frac{1}{3}\frac{\kappa}{\max\left(\sqrt{\kappa}, 1\right)} \geq \frac{\sqrt{\kappa}}{3\sqrt{\gamma}}\min\left(\sqrt{\kappa}, 1\right).$$

The result follows from (5.29) and the above inequalities. □

Theorem 5.10 implies that when solving the perturbed problems (PD$_\lambda$), if $\epsilon(A, b_\lambda, c_\lambda)$ is sufficiently small, we can predict the optimal active set of (PD) before $\mu_\lambda$ gets so small that we can even obtain the strictly complementary partition of (PD$_\lambda$). To see an example when (5.28) is satisfied, see our remarks after Theorem 5.13.

**Remark.** In Theorem 5.10, we do not require the optimal active set of (PD$_\lambda$) to be the same as the optimal active set of (PD). In fact, we will show that, in the numerical tests for the randomly generate problems (degenerate or nondegenerate), the optimal active sets of most perturbed problems are different from those of the original problems, but we can still predict sooner/better for (PD). In particular, the numerical experiments show that we are not solving (PD$_\lambda$) to high accuracy and there are iterations where we can predict the active set for (PD) but we are not close to the solution set of (PD$_\lambda$), nor able to predict the active set of (PD$_\lambda$); see page 81. □

### 5.3.2 Comparing with active-set prediction for (PD)

Similarly to Lemma 5.9, when we solve the original (PD) problems we can predict the optimal (PD) active set when the (PD) duality gap is smaller than some threshold. In this section, we intend to compare this threshold with the threshold value of $\mu_\lambda$ when we are able to predict the optimal active set of (PD) by solving (PD$_\lambda$) and show that the latter could be greater than the former under certain conditions (Theorem 5.13).

Lemma 5.13 in [135] — restated in this thesis as Lemmas 2.8 and 2.9 — yields an active-set prediction result for (PD). In fact this result can also be obtained by setting $\lambda = 0$ in Lemma 5.9, but for clarity, we restate it here.

**Lemma 5.11. [135, Lemma 5.13]** *Let $(x, y, s)$ in $\mathcal{N}_{-\infty}(\gamma)$, where $\mathcal{N}_{-\infty}(\gamma)$ is the neighbourhood defined in (2.10), and let $\mu$ as in (2.20). Let the cut-off value $C$ in (5.12) be set to $C = \frac{\epsilon(A,b,c)\gamma}{n}$, where $\epsilon(A, b, c)$ is defined in (2.22). When $\mu < \mu^{\max}$, where*

$$\mu^{\max} = \frac{\epsilon^2(A, b, c)}{n^2}\gamma, \tag{5.30}$$

*then for any strictly complementary solution $(x^*, y^*, s^*)$ of (PD) we have*

$$\bar{\mathcal{A}}(x) = \mathcal{A}(x^*),$$

*where $\bar{\mathcal{A}}(x)$ is defined in (5.12) and $\mathcal{A}(x^*)$ in (2.6).*

Before we deduce a relationship between $\mu_\lambda^{\max}$ in (5.26) and $\mu^{\max}$ in (5.30), we first relate two other important quantities, $\epsilon(A, b_\lambda, c_\lambda)$ and $\epsilon(A, b, c)$.

**Lemma 5.12.** *Assume (2.9) holds and (PD) has a unique and nondegenerate solution $(x^*, y^*, s^*)$. Then there exists at least one vector of perturbations $\bar{\lambda}(A, b, c, x^*, s^*) > 0$*

*such that for all* $0 \leq \lambda = \alpha\bar{\lambda} < \bar{\lambda}$ *where* $\alpha \in (0,1)$,

$$\epsilon(A, b_\lambda, c_\lambda) > \epsilon(A, b, c), \tag{5.31}$$

*where* $\epsilon(A, b_\lambda, c_\lambda)$ *is defined in* (5.2) *and* $\epsilon(A, b, c)$ *in* (2.22).

The proof of this lemma is given in the next section, Section 5.4.

**Theorem 5.13.** *In the conditions of Theorem 5.7, assume* (2.9) *holds and* (PD) *has a unique and nondegenerate solution* $(x^*, y^*, s^*)$. *Provided*

$$\epsilon(A, b, c) \leq \mathcal{O}\left(\sqrt{n\kappa}\min\left(\sqrt{\kappa}, 1\right)\right), \tag{5.32}$$

*where* $\kappa$ *is defined in* (5.27), *there exists at least one vector of perturbations* $\bar{\lambda}(A, b, c, x^*, s^*) > 0$ *such that*

$$\mu_\lambda^{\max} > \mu^{\max},$$

*for all* $0 < \lambda = \alpha\bar{\lambda} < \bar{\lambda}$, *where* $\alpha \in (0,1)$ *and where* $\mu_\lambda^{\max}$ *is defined in* (5.26) *and* $\mu^{\max}$ *in* (5.30).

**Proof.** Applying Theorem 5.10 with $\lambda = 0$ and so replacing $\epsilon(A, b_\lambda, c_\lambda)$ with $\epsilon(A, b, c)$, we deduce

$$\mu^{\max} < \frac{\kappa\epsilon(A, b, c)}{4n\left(\sqrt{n} + \epsilon(A, b, c)\right)}.$$

From Lemma 5.12, we have $\epsilon(A, b_\lambda, c_\lambda) > \epsilon(A, b, c)$. This and the definition of $\mu_\lambda^{\max}$ in (5.26) give

$$\frac{\kappa\epsilon(A, b, c)}{4n\left(\sqrt{n} + \epsilon(A, b, c)\right)} < \mu_\lambda^{\max}.$$

□

Theorem 5.13 implies that if $\epsilon(A, b, c)$ is sufficiently small, we may find the optimal active set of (PD) 'sooner' if we solve ($\text{PD}_\lambda$) using a primal-dual path-following IPM than if we solve (PD).

**Remark.** When (PD) has a unique solution $(x^*, y^*, s^*)$, we have

$$\epsilon(A, b, c) = \min\left(\min_{i \in \mathcal{I}} x_i^*,\ \min_{i \in \mathcal{A}} s_i^*\right) \leq \min_{i \in \mathcal{I}} x_i^* = \psi_p.$$

Note that according to [85] $\tau_p, \tau_d = \mathcal{O}(1)$ numerically. Thus provided $\psi_p > 1$ or $n$ is sufficiently large, (5.32) is satisfied. We illustrate this in an example next.

### 5.3.3 A simple example of predicting the optimal (PD) active set using perturbations

To illustrate our results, consider the following simple example

$$\min\ x_1 + 2x_2 \quad \text{subject to} \quad x_1 + x_2 = 1,\ x_1 \geq 0,\ x_2 \geq 0, \tag{5.33}$$

with the optimal solution

$$x^* = (1, 0), \quad \text{and} \quad y^* = 1,\ s^* = (0, 1).$$

Thus (5.33) has a unique and primal-dual nondegenerate solution with optimal active set $\mathcal{A}(x^*) = \{2\}$, and so $\psi_p = \epsilon(A, b, c) = 1$. Let the vector of perturbations be $\lambda = \alpha(1, 5)$ where $\alpha = 10^{-2}$. The perturbed problems ($\text{PD}_\lambda$) also have a unique solution

$$x^*_\lambda = (1 + 5\alpha, -5\alpha), \quad \text{and} \quad y^*_\lambda = 1 + \alpha,\ s^*_\lambda = (-\alpha, 1 - \alpha).$$

So $\epsilon(A, b_\lambda, c_\lambda) = \min(1 + 6\alpha, 1 + 4\alpha) = 1 + 4\alpha = 1.04$.

First we verify the conditions in Theorem 5.7, which are needed in both Theorems 5.10 and 5.13. Since it is not clear how to deduce the value of $\tau_p$ and $\tau_d$, we estimate them numerically[11] and it turns out that $\tau_p \approx \tau_d \approx 0.8$. We set the cut-off constant $C$ that separates the active and inactive constraints to be $C = \frac{\psi_p}{2} = 0.5$ and verify that

$$\|\lambda\| = \sqrt{26}\alpha < \frac{\psi_p}{16 \max(\tau_p, \tau_d)} < 1.$$

Thus the conditions in (5.18) are satisfied. Based on Theorem 5.7, we can predict the original optimal active set when $\mu_\lambda$ is less than $\mu^{\max}_\lambda \approx 0.0662$.

Next we verify Theorems 5.10 and 5.13. From (5.27), we get $\kappa \approx 1.25$, and so

$$\sqrt{n\kappa} \min\left(\sqrt{\kappa}, 1\right) \approx 1.58.$$

Thus

$$0 < \epsilon(A, b, c) < \epsilon(A, b_\lambda, c_\lambda) < \sqrt{n\kappa} \min\left(\sqrt{\kappa}, 1\right),$$

which implies that conditions (5.28) and (5.32) are satisfied. For the constant $\gamma$, it is common to choose a small value to have a large neighbourhood of the central path; set $\gamma = 0.01$. Then from (5.24) and (5.30), we have

$$\bar{\mu}^{\max}_\lambda \approx 0.0027 < \mu^{\max}_\lambda \quad \text{and} \quad \mu^{max} = 0.0025 < \mu^{\max}_\lambda.$$

This implies that when use perturbations, we can predict the original optimal active set sooner than the perturbed active set or the original active set without perturba-

[11] We estimate $\tau_p$ and $\tau_d$ from their definition in (ER), namely, we solve the following optimisation problem in MATLAB, $\max \|x - x^*\|/(r(x, s) + w(x, s))$ subject to $(x, y, s) \in \mathcal{F}^0_\lambda$, where $r(x, s)$ and $w(x, s)$ are defined in (5.1) and $\mathcal{F}^0_\lambda$ in (4.1); similarly for $\tau_d$.

tions. Furthermore, the threshold values (constant $C$) needed to separate the active constraints from the inactive ones for predicting the perturbed active set and the original active set without perturbations are 0.0052 and 0.005 respectively, both of which are much smaller than the cut-off $C = \frac{\psi_p}{2} = 0.5$ for predicting the original optimal active set using perturbations.

## 5.4 Proof of Lemma 5.12

Theorem 4.7 shows that we are able to preserve the optimal strict complementarity partition after perturbing the problems if the original (PD) has a unique and nondegenerate solution. Actually, we can take a step further and show that then ($PD_\lambda$) will also have a unique and nondegenerate solution.

**Theorem 5.14.** *Assume (2.9) holds and the (PD) problems have a unique and nondegenerate solution $(x^*, y^*, s^*)$. Let $\mathcal{A}$ and $\mathcal{I}$ denote the corresponding optimal active and inactive sets. Then there exists $\hat{\lambda} = \hat{\lambda}(A, b, c, x^*, s^*) > 0$ such that the perturbed problems ($PD_\lambda$) with $0 \leq \|\lambda\| < \hat{\lambda}$ have a unique and nondegenerate solution and the optimal active set is the same as that of the original (PD) problems.*

**Proof.** We consider the equivalent perturbed problem (4.6). From Theorem 4.7, we know there exists a $\hat{\lambda}(A, b, c, x^*, s^*) > 0$ such that (4.6) with $0 \leq \|\lambda\| < \hat{\lambda}$ has a strictly complementary solution $(\hat{p}, \hat{y}, \hat{q})$ with the same optimal active and inactive sets $\mathcal{A}$ and $\mathcal{I}$, namely we have

$$\hat{p}_\mathcal{I} > 0, \quad \hat{p}_\mathcal{A} = 0, \quad \hat{q}_\mathcal{A} > 0, \quad \text{and} \quad \hat{q}_\mathcal{I} = 0,$$

and also

$$A_\mathcal{I} \hat{p}_\mathcal{I} = b_\lambda. \tag{5.34}$$

Next we are about to show that $(\hat{p}, \hat{y}, \hat{q})$ is the unique solution of (4.6). Assume there exists another solution $\bar{p} \neq \hat{p}$. Then $(\bar{p}, \hat{y}, \hat{q})$ satisfies the optimality conditions (4.7). From the complementarity equations (the third term) in (4.7) and $\hat{q}_\mathcal{A} > 0$ we have $\bar{p}_\mathcal{A} = 0 = \hat{p}_\mathcal{A}$. Then we have $A_\mathcal{I} \bar{p}_\mathcal{I} = b_\lambda$. It follows from this and (5.34) that $A_\mathcal{I}(\bar{p}_\mathcal{I} - \hat{p}_\mathcal{I}) = 0$. As the (PD) solution is unique and nondegenerate, we must have $|\mathcal{I}| = m$ and $rank(A_\mathcal{I}) = m$, namely, $A_\mathcal{I}$ is nonsingular, which implies $\hat{p}_\mathcal{I} = \bar{p}_\mathcal{I}$. Then (4.6) has a unique and nondegenerate primal solution, which also implies unique and nondegenerate dual solution. □

To prove Lemma 5.12, we also need the following series of useful lemmas.

**Lemma 5.15 (Farkas' Lemma [10, Lemma 5.1]).** *One and only one of the follow-*

*ing two systems has a solution:*

$$\begin{aligned} &\textit{System 1:} \quad Tw \geq 0 \quad \textit{and} \quad b^T w < 0, \\ &\textit{System 2:} \quad T^T y = b \quad \textit{and} \quad y \geq 0, \end{aligned}$$

*where* $T \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $w \in \mathbb{R}^n$ *and* $y \in \mathbb{R}^m$.

**Lemma 5.16.** *Given* $i \in \{1, \ldots, n\}$, *the following system*

$$\begin{cases} y + Ax \geq 0 \\ x - A^T y \geq 0 \quad \textit{and} \quad x_i - A_i^T y > 0 \\ (x, y) \geq 0 \end{cases}$$

*always has a solution, where* $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ *and* $A_i$ *is the* $i^{th}$ *column of* $A$.

**Proof.** Without losing generality, we can choose $i = 1$. Partition $x$ and $A$ as $x = \begin{bmatrix} x_1 & \bar{x}^T \end{bmatrix}^T$ and $A = \begin{bmatrix} A_1 & \bar{A} \end{bmatrix}$, where $\bar{x} = \begin{bmatrix} x_2 & \ldots & x_n \end{bmatrix}^T$ and $\bar{A} = \begin{bmatrix} A_2 & \ldots & A_n \end{bmatrix}$.

We need to prove the following system has a solution

$$\begin{cases} \begin{array}{rcrcrcl} y & + & A_1 x_1 & + & \bar{A}\bar{x} & \geq & 0 \\ -\bar{A}^T y & & & + & \bar{x} & \geq & 0 \\ y & & & & & \geq & 0 \\ & & x_1 & & & \geq & 0 \\ & & & & \bar{x} & \geq & 0 \\ A_1^T y & - & x_1 & & & < & 0 \end{array} \end{cases} . \tag{5.35}$$

From Lemma 5.15, we know (5.35) has a solution if and only if

$$\begin{cases} \begin{bmatrix} I_m & -\bar{A} & I_m & 0 & 0 \\ A_1^T & 0 & 0 & 1 & 0 \\ \bar{A}^T & I_{n-1} & 0 & 0 & I_{n-1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} A_1 \\ -1 \\ 0 \end{bmatrix} , \\ (u_1, u_2, u_3, u_4, u_5) \geq 0 \end{cases} \tag{5.36}$$

has no solution, where $u_1 \in \mathbb{R}^m$, $u_2 \in \mathbb{R}^{n-1}$, $u_3 \in \mathbb{R}^m$, $u_4 \in \mathbb{R}$ and $u_5 \in \mathbb{R}^{n-1}$.

Assume (5.36) has a solution $(u_1, u_2, u_3, u_4, u_5) \geq 0$. Then we get

$$u_1 - \bar{A}u_2 - A_1 = -u_3 \leq 0, \tag{5.37a}$$

$$A_1^T u_1 = -1 - u_4 < 0, \tag{5.37b}$$

$$\bar{A}^T u_1 = -u_2 - u_5 \leq 0. \tag{5.37c}$$

Multiplying both sides of (5.37a) by $u_1^T \geq 0$, we have

$$u_1^T u_1 - (\bar{A}^T u_1)^T u_2 - A_1^T u_1 = -u_1^T u_3.$$

From (5.37b), (5.37c) and the nonnegativity of the variables, we know

$$u_1^T u_1 - (\bar{A}^T u_1)^T u_2 - \bar{A}_1^T u_1 > 0 \quad \text{but} \quad -u_1^T u_3 \leq 0.$$

Thus (5.36) has no solution, which implies (5.35) has a solution. □

**Lemma 5.17.** *The system*

$$\begin{cases} y + Ax \geq 0 \\ x - A^T y > 0 \\ (x, y) \geq 0 \end{cases} \tag{5.38}$$

*always has a solution, where $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$.*

**Proof.** From Lemma 5.16, we know for any $i \in \{1, \ldots, n\}$, there exists $(x^i, y^i) \geq 0$ where $x^i \in \mathbb{R}^n$ and $y^i \in \mathbb{R}^m$, such that

$$\begin{cases} y^i + Ax^i \geq 0, \\ x^i - A^T y^i \geq 0 \quad \text{and} \quad x_i^i - A_i^T y^i > 0. \end{cases} \tag{5.39}$$

Set $\hat{x} = \sum_{i=1}^n x^i \geq 0$ and $\hat{y} = \sum_{i=1}^n y^i \geq 0$. Then from (5.39), we have

$$\hat{y} + A\hat{x} = \sum_{i=1}^n y^i + A(\sum_i^n x^i) = \sum_{i=1}^n (y^i + Ax^i) \geq 0,$$

and

$$\hat{x} - A^T \hat{y} = \sum_{i=1}^n (x^i - A^T y^i) = \begin{bmatrix} (x_1^1 - A_1^T y^1) + (x_1^2 - A_1^T y^2) + \cdots + (x_1^n - A_1^T y^n) \\ \vdots \\ (x_n^1 - A_n^T y^1) + (x_n^2 - A_n^T y^2) + \cdots + (x_n^n - A_n^T y^n) \end{bmatrix} > 0.$$

□

**Lemma 5.18.** *The system*

$$\begin{cases} y + Ax > 0 \\ x - A^T y \geq 0 \\ (x, y) \geq 0 \end{cases} \tag{5.40}$$

*always has a solution, where $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$.*

**Proof.** Replace A in Lemma 5.17 by $-A^T$. □

**Lemma 5.19.** *The system*

$$\begin{cases} y + Ax > 0 \\ x - A^T y > 0 \\ (x, y) \geq 0. \end{cases}$$

*always has a solution, where* $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$ *and* $y \in \mathbb{R}^m$.

**Proof.** From Lemmas 5.17 and 5.18, we know there exist $(\hat{x}, \hat{y})$ and $(\tilde{x}, \tilde{y})$ such that (5.38) and (5.40) hold respectively. Set $\bar{x} = \hat{x} + \tilde{x}$ and $\bar{y} = \hat{y} + \tilde{y}$ and deduce

$$\bar{y} + A\bar{x} = \underbrace{(\hat{y} + A\hat{x})}_{\geq 0} + \underbrace{(\tilde{y} + A\tilde{x})}_{>0} > 0 \quad \text{and} \quad \bar{x} - A^T\bar{y} = \underbrace{(\hat{x} - A^T\hat{y})}_{>0} + \underbrace{(\tilde{x} - A^T\tilde{y})}_{\geq 0} > 0.$$

□

**Proof of Lemma 5.12.** Assume $\mathcal{A}$ and $\mathcal{I}$ are the optimal active and inactive sets at the unique solution $(x^*, y^*, s^*)$ of (PD). Then from (2.22), we have

$$\epsilon(A, b, c) = \min\left(\min_{i \in \mathcal{I}}(x_i^*), \min_{i \in \mathcal{A}}(s_i^*)\right). \tag{5.41}$$

From Theorem 5.14, we know there exists a $\hat{\lambda} = \hat{\lambda}(A, b, c, x^*, s^*)$ such that ($\text{PD}_\lambda$) with $0 \leq \|\lambda\| < \hat{\lambda}$ has a unique and nondegenerate solution and $\mathcal{A}$ and $\mathcal{I}$ are the optimal active and inactive sets. Since $(\hat{p}, \hat{y}, \hat{q})$ defined in (4.14) is a solution of (4.6), $(x_\lambda^*, y_\lambda^*, s_\lambda^*) = (\hat{p} - \lambda, \hat{y}, \hat{q} - \lambda)$ is a solution of ($\text{PD}_\lambda$) and also unique, with $\mathcal{A}$ and $\mathcal{I}$ being the optimal active and inactive sets. This and (5.2) give

$$\epsilon(A, b_\lambda, c_\lambda) = \min\left(\min_{i \in \mathcal{I}}(\hat{x}_i + \lambda_i), \min_{i \in \mathcal{A}}(\hat{s}_i + \lambda_i)\right). \tag{5.42}$$

From (4.14), recalling that $\hat{p} = \hat{x} + \lambda$ and $\hat{q} = \hat{s} + \lambda$, we have

$$(\hat{x}_\mathcal{I} + \lambda_\mathcal{I}) - x_\mathcal{I}^* = \lambda_\mathcal{I} + A_\mathcal{I}^{-1}A_\mathcal{A}\lambda_\mathcal{A} \quad \text{and} \quad (\hat{s}_\mathcal{A} + \lambda_\mathcal{A}) - s_\mathcal{A}^* = \lambda_\mathcal{A} - (A_\mathcal{I}^{-1}A_\mathcal{A})^T\lambda_\mathcal{I}.$$

This, (5.41) and (5.42) give us that $\epsilon(A, b_\lambda, c_\lambda) > \epsilon(A, b, c)$, provided

$$\begin{cases} \lambda_\mathcal{I} + A_\mathcal{I}^{-1}A_\mathcal{A}\lambda_\mathcal{A} > 0 \\ \lambda_\mathcal{A} - (A_\mathcal{I}^{-1}A_\mathcal{A})^T\lambda_\mathcal{I} > 0 \\ \lambda_\mathcal{I}, \lambda_\mathcal{A} \geq 0 \end{cases}. \tag{5.43}$$

It remains to find a solution of (5.43) whose norm is less than $\hat{\lambda}$. From Lemma 5.19, we know (5.43) always has a solution, say $\bar{\lambda}$. Since (5.43) is homogeneous, $\frac{\hat{\lambda}}{2\|\bar{\lambda}\|}\bar{\lambda}$ is also a solution, and $\|\frac{\hat{\lambda}}{2\|\bar{\lambda}\|}\bar{\lambda}\| < \hat{\lambda}$. Without losing generality, we denote this solution as $\bar{\lambda}$. Furthermore, (5.43) holds for all $\lambda$ with $0 < \lambda = \alpha\bar{\lambda} < \bar{\lambda}$ where $\alpha \in (0, 1)$. □

## 5.5 Using the identification function as threshold in the controlled perturbations prediction framework

The purpose of this section is to illustrate that the idea of using controlled perturbations serves as a general framework for active-set prediction for IPMs that has the potential to work with other prediction strategies, not just cut-off. We investigate the use of the identification function (Section 3.1) in this context.

In Section 3.1 we have introduced the so-called identification function and formulated such a function for LP problems. Recall the identification function $\rho(x, y, s)$ defined in (3.1).

Let $(x, y, s) \in \mathcal{F}_\lambda^0$, where $\mathcal{F}_\lambda^0$ is defined in (4.1), namely, $(x, y, s)$ is a strictly feasible point of ($\text{PD}_\lambda$), for some $\lambda \geq 0$. Then we can deduce the following 'feasible' variant of the identification function,

$$\rho(x, y, s) = \sqrt{r(x, s) + w(x, s)}, \tag{5.44}$$

where $r(x, s)$ and $w(x, s)$ are defined in (5.1).

**Lemma 5.20.** *Let $(x, y, s) \in \mathcal{F}_\lambda^0$, where $\mathcal{F}_\lambda^0$ is defined in (4.1) for some $\lambda \geq 0$. Then*

$$\rho^2(x, y, s) \leq C_2\mu_\lambda + 4\|\lambda\| \max(\|\lambda\|, 1), \tag{5.45}$$

*where $\rho(x, y, s)$ is defined in (5.44), $\mu_\lambda$ in (5.4) and $C_2 > 0$ in (5.11).*

**Proof.** (5.45) follows from (5.44), Lemmas 5.3 and 5.4. □

A threshold test is employed to predict the active and strongly active sets. Recall that $\mathcal{A}(x^*)$ in (2.6a) and $\mathcal{A}^+(s^*)$ in (2.6b) represent the active and strongly active sets at a (PD) solution $(x^*, y^*, s^*)$. We have also denoted $\hat{\mathcal{A}}(x, y, s)$ in (3.3) and $\hat{\mathcal{A}}^+(x, y, s)$ in (3.4) as the predicted active and strongly active sets when using the identification function, respectively. The following theorem shows that $\hat{\mathcal{A}}(x, y, s)$ and $\hat{\mathcal{A}}^+(x, y, s)$ bound well the optimal active and strongly active sets of the original (PD) problems.

**Theorem 5.21.** *Fix the vector of perturbations $\lambda$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{1}{8\max(\tau_p^2, \tau_d^2)}\right), \tag{5.46}$$

*where $\tau_p$ and $\tau_d$ are the problem-dependent constants in (5.10). Let $(x, y, s) \in \mathcal{F}_\lambda^0$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \frac{1}{2C_2\max(\tau_p^2, \tau_d^2)}, \tag{5.47}$$

*where $\mathcal{F}_\lambda^0$ is defined in (4.1), $\mu_\lambda$ in (5.4) and $C_2 > 0$ in (5.11) is a problem-dependent constant when $\lambda$ is fixed. Then there exists a (PD) solution $(x^*, y^*, s^*)$ such that*

$$\hat{\mathcal{A}}^+(x, y, s) \subseteq \mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x, y, s),$$

*where $\hat{\mathcal{A}}(x, y, s)$ is defined in (3.3), $\hat{\mathcal{A}}^+(x, y, s)$ in (3.4), $\mathcal{A}(x^*)$ in (2.6a), and $\mathcal{A}^+(s^*)$ in (2.6b).*

**Proof.** From Lemma 3.3 and norm properties, we have

$$x_i^* - \tau_p \rho^2(x, y, s) \leq x_i \leq x_i^* + \tau_p \rho^2(x, y, s) \tag{5.48}$$

and

$$s_i^* - \tau_d \rho^2(x, y, s) \leq s_i \leq s_i^* + \tau_d \rho^2(x, y, s). \tag{5.49}$$

From (5.45), (5.46) and (5.47), we have $\rho^2(x, y, s) < \frac{1}{\max(\tau_p^2, \tau_d^2)}$, and so,

$$\rho(x, y, s) < \frac{1}{\max(\tau_p, \tau_d)}. \tag{5.50}$$

If $i \in \mathcal{A}(x^*)$, $x_i^* = 0$. Then the second inequality in (5.48), together with (5.50) give us that

$$x_i \leq \tau_p \rho^2(x, y, s) < \frac{\tau_p}{\max(\tau_p, \tau_d)} \rho(x, y, s) \leq \rho(x, y, s).$$

Thus $i \in \hat{\mathcal{A}}(x, y, s)$, and $\mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x, y, s)$. If $i \notin \mathcal{A}^+(s^*)$, $s_i^* = 0$. Then from (5.49) and (5.50), we have

$$s_i \leq \tau_d \rho^2(x, y, s) < \rho(x, y, s).$$

Thus $i \notin \hat{\mathcal{A}}^+(x, y, s)$, namely $\hat{\mathcal{A}}^+(x, y, s) \subseteq \mathcal{A}^+(s^*)$. From the complementary conditions (2.3c), we have $\mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*)$. □

Similarly to Theorems 5.7 and 5.8, we can also prove that under certain conditions the predicted (strongly) active set for (PD) coincides with the actual optimal (strongly) active set of (PD), when using the identification function for prediction; see the following Theorems 5.22 and 5.23.

**Theorem 5.22.** *Assume $\psi_p > 0$, where $\psi_p$ is defined in (5.17). Fix $\lambda$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{1}{8\max(\tau_p^2, \tau_d^2)}, \frac{\psi_p^2}{32}\right) \tag{5.51}$$

*where $\tau_p$ and $\tau_d$ are the problem-dependent constants defined in (5.10). Let*

*$(x, y, s) \in \mathcal{F}_\lambda^0$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \min\left(\frac{1}{2C_2 \max(\tau_p^2, \tau_d^2)}, \frac{\psi_p^2}{8C_2}\right), \tag{5.52}$$

*where $\mathcal{F}_\lambda^0$ is defined in (4.1), $\mu_\lambda$ in (5.4) and $C_2 > 0$ in (5.11). Then there exists a (PD) solution $(x^*, y^*, s^*)$ such that*

$$\hat{\mathcal{A}}(x, y, s) = \mathcal{A}(x^*),$$

*where $\hat{\mathcal{A}}(x, y, s)$ is defined in (5.12) and $\mathcal{A}(x^*)$ in (2.6a).*

**Proof.** From Theorem 5.21, we have $\mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x, y, s)$. It remains to prove $\hat{\mathcal{A}}(x, y, s) \subseteq \mathcal{A}(x^*)$. From (5.45), (5.51) and (5.52), we have

$$\rho(x, y, s) < \frac{\psi_p}{2}. \tag{5.53}$$

If $i \notin \mathcal{A}(x^*)$, from the first inequality in (5.48), (5.50) and (5.53), we have

$$x_i \geq x_i^* - \tau_p \rho^2(x, y, s) > \psi_p - \rho(x, y, s) > \rho(x, y, s).$$

Thus $i \notin \hat{\mathcal{A}}^+(x, y, s)$, namely $\hat{\mathcal{A}}(x, y, s) \subseteq \mathcal{A}(x^*)$. □

**Theorem 5.23.** *Assume $\psi_d > 0$, where $\psi_d$ is defined in (5.20). Fix $\lambda$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{1}{8 \max(\tau_p^2, \tau_d^2)}, \frac{\psi_d^2}{32}\right) \tag{5.54}$$

*where $\tau_p$ and $\tau_d$ are the problem-dependent constants defined in (5.10). Let $(x, y, s) \in \mathcal{F}_\lambda^0$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \min\left(\frac{1}{2C_2 \max(\tau_p^2, \tau_d^2)}, \frac{\psi_d^2}{8C_2}\right), \tag{5.55}$$

*where $\mathcal{F}_\lambda^0$ is defined in (4.1), $\mu_\lambda$ in (5.4) and $C_2 > 0$ in (5.11). Then there exists a (PD) solution $(x^*, y^*, s^*)$ such that*

$$\hat{\mathcal{A}}^+(x, y, s) = \mathcal{A}^+(s^*),$$

*where $\hat{\mathcal{A}}^+(x, y, s)$ is defined in (5.12) and $\mathcal{A}^+(s^*)$ in (2.6b).*

**Proof.** From Theorem 5.21, we have $\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*)$. It remains to prove $\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*)$. From (5.45), (5.54) and (5.55), we have

$$\rho(x,y,s) < \frac{\psi_d}{2}. \tag{5.56}$$

Similarly to the proof of Theorem 5.22, if $i \in \mathcal{A}^+(s^*)$, from (5.48), (5.50) and (5.56), we have $s_i > \rho(x,y,s)$, which implies $\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*)$. □

Note that the assumptions on $\psi_p$ and $\psi_d$ are not required in implementations; see Section 6.4 for numerical experiments combining the identification function and controlled perturbations.

**Comparing perturbed and unperturbed active-set predictions using the identification function as threshold.** Similarly to Theorems 5.21 – 5.23, when we solve the original (PD) problems we can also predict the optimal (PD) active set using the identification function when the (PD) duality gap $\mu$ is smaller than some threshold. Letting $\lambda = 0$ and following the proofs of Lemmas 5.3 and 5.4, we can derive an upper bound for $\rho(x,y,s)$ (5.44), namely,

$$\rho^2(x,y,s) \leq \bar{C}_2 \mu, \tag{5.57}$$

where $\mu$ is defined in (2.20), $\epsilon(A,b,c)$ in (2.22) and

$$\bar{C}_2 = \frac{n\sqrt{n}}{\epsilon(A,b,c)} + n. \tag{5.58}$$

Using (5.57) instead of (5.45), and following the same procedure of the proofs of Theorems 5.21 – 5.23, we can deduce similar results as in these theorems for (PD) by setting $\lambda = 0$. Namely, let $(x,y,s) \in \mathcal{F}^0$ with $\mu$ sufficiently small,

$$\mu < \frac{1}{\bar{C}_2 \max(\tau_p^2, \tau_d^2)}.$$

Then there exists a (PD) solution $(x^*,y^*,s^*)$ such that $\hat{\mathcal{A}}(x,y,s)$ and $\hat{\mathcal{A}}^+(x,y,s)$ bound well the optimal active and strongly active sets of (PD) problems, namely,

$$\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x,y,s),$$

where $\hat{\mathcal{A}}(x,y,s)$ is defined in (3.3), $\hat{\mathcal{A}}^+(x,y,s)$ in (3.4), $\mathcal{A}(x^*)$ in (2.6a), and $\mathcal{A}^+(s^*)$ in (2.6b). Further, if $\psi_p > 0$ and

$$\mu < \min\left(\frac{1}{\bar{C}_2 \max(\tau_p^2, \tau_d^2)}, \frac{\psi_p^2}{4\bar{C}_2}\right), \tag{5.59}$$

then there exists a (PD) solution $(x^*, y^*, s^*)$ such that $\hat{\mathcal{A}}(x, y, s) = \mathcal{A}(x^*)$, where $\psi_p$ is defined in (5.17); if $\psi_d > 0$ and

$$\mu < \min\left(\frac{1}{\bar{C}_2 \max(\tau_p^2, \tau_d^2)}, \frac{\psi_d^2}{4\bar{C}_2}\right), \tag{5.60}$$

then there exists a (PD) solution $(x^*, y^*, s^*)$ such that $\hat{\mathcal{A}}^+(x, y, s) = \mathcal{A}^+(s^*)$, where $\psi_d$ is defined in (5.20).

**Remark.** To find the conditions when we can predict the optimal (PD) active set 'sooner' if we solve ($\text{PD}_\lambda$) using a primal-dual path-following IPM with the identification function than if we solve (PD), we compare the threshold (upper bound) for $\mu_\lambda$ in (5.52) with that for $\mu$ in (5.59), as well as by comparing (5.55) with (5.60). Essentially we need

$$\frac{1}{2C_2} > \frac{1}{\bar{C}_2},$$

which is equivalent to

$$\epsilon(A, b, c)\epsilon(A, b_\lambda, c_\lambda) + \sqrt{n}\,(2\epsilon(A, b, c) - \epsilon(A, b_\lambda, c_\lambda)) < 0. \tag{5.61}$$

If $n$ is large and $\epsilon(A, b_\lambda, c_\lambda)$ is at least twice as large as $\epsilon(A, b, c)$, (5.61) may hold. However, in theory, it is not trivial to derive a general condition for this to hold. □

## 5.6 Conclusions

To tackle the challenging question of early optimal active-set predictions for IPMs, we have proposed the use of controlled perturbations for improving these capabilities of IPMs for LP. The perturbations are chosen so as to slightly enlarge the feasible set in the hope that the central path of the perturbed problems passes through or close to the original solution set when the perturbed barrier parameter is not too small. Our approach solves a (sequence of) perturbed problems using a standard primal-dual path-following method and predicts using cut-off, the optimal active set of the original problem on the way.

We also illustrated that the idea of controlled perturbations can be used with different active-set prediction strategies, such as the identification function.

Please refer to Chapter 6 for preliminary numerical experiments.

# 6 Numerical Experiments for Active-set Prediction Using Perturbations

In this chapter, we present the implementation details of the perturbed algorithm (Section 6.1) and our preliminary numerical experiments. We carry out two types of tests, one comparing the accuracy of the predicted active sets and the other one exploring the case of crossover to the simplex method. We first conduct the tests using cut-off in the threshold test to predict the active constraints (Section 6.3). For verifying the accuracy of our active-set predictions, we apply an infeasible primal-dual path-following IPM to perturbed and original randomly-generated LP problems, terminate the algorithm at various iterations and compare the accuracy of predictions using certain correctness comparison ratios (Section 6.3.1). When crossing over to the simplex method, we test the efficiency of our active-set predictions by comparing the number of simplex iterations needed to solve the original problem to optimality, after some initial IPM iterations (Section 6.3.2). Then we replace the cut-off procedure with the identification function and repeat all the tests in Section 6.4. Comparisons between the perturbed algorithm with cut-off and with the identification function are also presented at the end of this section.

## 6.1 The perturbed algorithm and implementation

All numerical experiments in this section employ an infeasible primal-dual path-following interior point method applied to ($PD_\lambda$) or (PD). The perturbed algorithm is summarised in **Algorithm 6.1** and it is nothing but Algorithm 3.1 applied to ($PD_\lambda$) with possible shrinkage of the perturbations.

**Algorithm 6.1** The Perturbed Interior Point Algorithm with Active-set Prediction

**Step 0**: choose perturbations $(\lambda^0, \phi^0) > 0$ and calculate a Mehrotra starting point $(x^0, y^0, s^0)$ according to Section 2.2.2 on page 22;

**for** $k = 0, 1, 2, \ldots$ **do**

**Step 1:** solve the perturbed system (4.8) by the augmented system approach, namely

$$\begin{bmatrix} -D_\lambda^{-2} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \end{bmatrix} = - \begin{bmatrix} R_d^k - (X^k + \Lambda^k)^{-1} R_{\mu_\lambda}^k \\ R_p^k \end{bmatrix},$$
$$\Delta s^k = - \left(X^k + \Lambda^k\right)^{-1} \left(R_{\mu_\lambda}^k + \left(S^k + \Phi^k\right) \Delta x^k\right),$$

where $D_\lambda = \left(S^k + \Phi^k\right)^{-\frac{1}{2}} \left(X^k + \Lambda^k\right)^{\frac{1}{2}}$, $R_p^k = Ax^k - b$, $R_d^k = A^T y^k + s^k - c$, $R_{\mu_\lambda}^k = \left(X^k + \Lambda^k\right)\left(S^k + \Phi^k\right) e - \sigma^k \mu_\lambda^k e$, and where $\sigma^k \in [0, 1]$ is defined in (2.15) and

$$\mu_\lambda^k = \frac{(x^k + \lambda^k)^T (s^k + \phi^k)}{n}; \tag{6.1}$$

**Step 2:** choose a fixed, close to 1, fraction of the stepsize to the nearest constraint boundary in the primal and dual space, respectively. Namely,

$$\alpha_p^k = \min\left(\bar{\alpha} \min_{i:\Delta x_i^k < 0} \left(\frac{-x_i^k - \lambda_i^k}{\Delta x_i^k}\right), 1\right) \quad \text{and} \quad \alpha_d^k = \min\left(\bar{\alpha} \min_{i:\Delta s_i^k < 0} \left(\frac{-s_i^k - \phi_i^k}{\Delta s_i^k}\right), 1\right), \tag{6.2}$$

where $\bar{\alpha} = 0.9995$;

**Step 3:** update $x^{k+1} = x^k + \alpha_p^k \Delta x^k$ and $(y^{k+1}, s^{k+1}) = (y^k, s^k) + \alpha_d^k (\Delta y^k, \Delta s^k)$;

**Step 4:** predict the optimal active set of (PD) and denote by $\mathcal{A}^k$;

**Step 5:** terminate if some termination criterion is satisfied;

**Step 6:** calculate $(\lambda^{k+1}, \phi^{k+1})$ possibly by shrinking $(\lambda^k, \phi^k)$ so that $(x^{k+1} + \lambda^{k+1}, s^{k+1} + \phi^{k+1}) > 0$;

**end for**

**Algorithm without perturbations.** For comparison purposes, we use Algorithm 3.1 as the unperturbed algorithm. (It is the same as Algorithm 6.1 with $\lambda^k = \phi^k = 0$ and hence no Step 6 that calculates the changes to $(\lambda^k, \phi^k)$.)

**Preprocessing.** We use the same preprocessing code explained in Section 3.4.1. The main aim of this preprocessor is to check whether matrix $A$ has full row rank; if not, transform it to a full row rank matrix.

**Choice of initial perturbations.** In our theory, we have used the same vector of perturbations for both primal and dual variables. For better numerical efficiency, we have different perturbations $\lambda$ and $\phi$ for primal and dual variables respectively. We set the initial perturbations to be $\lambda^0 = \phi^0 = 10^{-2} e$, where $e$ is a vector of ones.

We have done experiments to explore the sensitivity of our algorithm to the value of the initial perturbations. For example, choosing $\lambda^0 = \phi^0 = 10^{-1}e$ yields a high false-prediction ratio (proportion of mistakes). Perturbations of order $10^{-2}$ and $10^{-3}$ yield quickly a good approximation of the original (PD) active set. For $\lambda^0 = \phi^0 = 10^{-4}e$, the perturbed algorithm starts to behave similarly to the unperturbed one simply because the perturbations are too small.

**Active-set prediction.** Recall the active-set prediction framework, Procedure 3.2, which moves the indices of the primal variables between three sets, the predicted active and inactive sets, and the undetermined set, during the run of an IPM algorithm. In this chapter, we integrate this strategy into Algorithm 6.1 to predict the active set of the original (PD). We employ Procedure 3.2 with cut-off for tests in Section 6.3. In Section 6.4, we discuss the efficiency of predicting the optimal active set of the original problems using the identification function (3.1) and so we employ the identification function as the threshold in Procedure 3.2. For implementation details of Procedure 3.2, cut-off and the identification function, please refer to Section 3.4.1 in Chapter 3.

Note that, the indicators [32] introduced in Chapter 3, however, are not suitable for our purposes because of their construction, as when employed in Algorithm 6.1, they can only be used to predict the optimal active set of ($PD_\lambda$), not that of (PD). Thus we will not conduct any tests using indicators.

**Termination.** There are different ways to terminate the algorithm.

- **The relative residual of the perturbed problem**. If it is very small, the algorithm may have solved the problem too far and so suffer from the ill-conditioning issue. By relative residual, we consider the following quantity

$$\text{relRes}_\lambda^k = \frac{|| \left(Ax^k - b, A^T y^k + s^k - c, \left(X^k + \Lambda^k\right)\left(S^k + \Phi^k\right) e\right) ||_\infty}{1 + \max\left(||b||_\infty, ||c||_\infty\right)}, \tag{6.3}$$

  where $\|\cdot\|_\infty$ represents the infinity-norm, namely $\|x\|_\infty = \max\limits_{i=1,\ldots,n}(|x_i|)$ and $x = (x_1, x_2, \ldots, x_n)$. When $\lambda^k = \phi^k = 0$, (6.3) is the same as (2.18).

- **The value of $\mu_\lambda$**. We can terminate the algorithm when $\mu_\lambda$ is less than a user-defined threshold, for instance $10^{-3}$. This would not be a wise choice if we intend to find the optimal solution of ($PD_\lambda$); our aim, however, is to predict a satisfying proportion of the optimal active set of (PD) early on.

- We could perform **crossover to the simplex method**. An initial basis can be generated to start the simplex method from the predicted active set of Algorithm 6.1 at low extra computational cost. A good prediction could yield substantial savings of simplex iterations, which can be seen from our preliminary numerical results.

The termination criteria for each test in the following sections can consist of one or more conditions from the above list, which will be stated clearly at the beginning of each test.

**Shrinking the perturbations.** In theory (Theorems 5.7 and 5.8), we state that under certain conditions on problem degeneracy, when the perturbations are sufficiently small, we are able to predict the optimal active set of the original problem by solving the perturbed problem. If we consider a sequence of vectors of perturbations $\{\lambda^k\}$, where $\lambda^{k+1} < \lambda^k$ for all $k = 0, 1, 2, \ldots$, the theoretical results still hold, namely we can still predict the optimal active set of the original problem, say at iteration $N$, provided $\lambda^N$ and $\mu_\lambda^N$ are sufficiently small. In practice, when the initial perturbations are not sufficiently effective (e.g. too large) for active-set prediction, we allow them to shrink after some IPM iterations so that the resulting perturbed feasible set is smaller but still contains the feasible set of the original problem. This enables the algorithm to adjust the perturbations accordingly, namely, the perturbed problems are getting 'closer' to the original one. Our numerical experience seems to imply that shrinking perturbations generally improves the performance of the perturbed algorithm, especially when there is no satisfactory heuristic algorithm for choosing initial perturbations. Assume $t^{k+1} = \min(x^{k+1})$ and $v^{k+1} = \min(s^{k+1})$. We update the perturbations as follows,

$$\lambda^{k+1} = \begin{cases} \eta\lambda^k, & \text{if } t^{k+1} > 0 \\ (1-\zeta)\lambda^k + \zeta(-t^{k+1})e, & \text{if } t^{k+1} \leq 0 \end{cases},$$

and

$$\phi^{k+1} = \begin{cases} \eta\phi^k, & \text{if } v^{k+1} > 0 \\ (1-\zeta)\phi^k + \zeta(-v^{k+1})e, & \text{if } v^{k+1} \leq 0 \end{cases},$$

where $\eta \in (0, 1]$ and $\zeta \in (0, 1)$. It follows that

$$x^{k+1} + \lambda^{k+1} > 0 \quad \text{and} \quad s^{k+1} + \phi^{k+1} > 0.$$

We observed in our numerical experiments that when solving nondegenerate problems, it is better to shrink faster, roughly keeping the perturbations to be $\mathcal{O}(\mu_\lambda)$. When solving degenerate problems however, it is better to shrink slower, at a rate of $\mathcal{O}(\sqrt{\mu_\lambda})$. It is difficult and often impossible to distinguish a priori between degenerate and non-degenerate cases. After several numerical trials, we chose to set $\eta = 1$ and $\zeta = 0.5$.

## 6.2 Test problems

We employ the same test problems introduced in Section 3.4.2, which we tested Algorithm 3.1 on. For clarity, we briefly review them in the following and include some useful remarks.

**Random problems (TS1 and TS2).** We employ two sets of randomly generated test problems, TS1 and TS2. Recall that the majority of the problems in TS1 are primal nondegenerate and dual degenerate, while TS2 is a set of randomly generated primal-dual degenerate LP test problems. There are 100 test problems in each test set. Whenever we use TS1 or TS2, we start from the same seed to make sure we test on the same problems throughout the thesis.

We now briefly address the following questions for test problems in both TS1 and TS2.

1. Is the actual active set of ($PD_\lambda$) different from that of (PD)? And if so, how different is it?

2. Do the perturbed problems have a unique solution? And if so, is it nondegenerate?

**On the difference between the active sets of ($PD_\lambda$) and (PD).** To answer the first set of questions, we first solve the perturbed and unperturbed problems to optimality using an interior point solver (IPM solver from LINPROG in MATLAB) and consider all variables less than $10^{-5}$ to be active. Then we find the optimal solutions of the above two problems using a simplex solver (simplex solver from LINPROG) and set all variables less than $10^{-5}$ as active as well. If the active set of ($PD_\lambda$) obtained by IPM is not the same as that of (PD) from IPM, or the active set of ($PD_\lambda$) obtained from the simplex solver is not the same as that of (PD) from the simplex solver, we consider that ($PD_\lambda$) and (PD) have different active sets. We found that the optimal active set of the perturbed problem is different from the original optimal active set for 98% of the test problems in TS1 and all test problems in TS2. Furthermore, for problems in TS1, the average difference between the strictly complementary partition of ($PD_\lambda$) and that of (PD) is as high as 33% and the difference between the active set at a vertex solution of ($PD_\lambda$) and that of (PD) is about 15% on average; for TS2, the average difference between the strictly complementary partitions of ($PD_\lambda$) and (PD) is about 29% and the difference between active sets at vertex solutions is 17% on average.

**On the uniqueness and degeneracy of the ($PD_\lambda$) solutions.** To attempt to answer the second set of questions, we check if the primal active set of the perturbed problem ($PD_\lambda$) at a strictly complementary solution is the same as that at a vertex solution. If so, we consider ($PD_\lambda$) has a unique primal solution. If this primal solution has more than $m$ (number of constraints) positive components (components greater than $10^{-5}$), we record that this solution is nondegenerate. From Theorem 2.5 we know if the primal problem has a unique and nondegenerate solution, so is the dual; thus ($PD_\lambda$) has a unique and primal-dual nondegenerate solution. The interesting observation is that, for both TS1 and TS2 over 95% of the perturbed problems have a unique and nondegenerate solution, regardless of the uniqueness or degeneracy of the original test problems.

**Netlib problems (TS3)** We have selected 37 small and medium-sized problems from the NETLIB test set, the same ones as in Section 3.4.2.

## 6.3 Numerical results using cut-off for active-set prediction

### 6.3.1 On the accuracy of active-set predictions using prediction ratios

In this section, we illustrate and discuss the benefits of using perturbations to predict the optimal active set.

The main task for this test is to compare the three measures, false-prediction, missed-prediction and correctness ratios, for Algorithms 6.1 and 3.1. We have introduced the three ratios in Section 3.4.3. Here we briefly restate their meaning. False-prediction ratio measures the degree of incorrectly identified active constraints, missed-prediction ratio measures the proportion of incorrectly rejected active constraints and correctness ratio shows the accuracy of the prediction. All three ratios range from 0 to 1. If the predicted set is the same as the actual optimal active set, then the correctness ratio is 1; for details of calculating these ratios, please refer to the corresponding paragraphs in Section 3.4.3.

When an LP problem has multiple solutions, the active set of a vertex solution is different from that of the strictly complementary solutions (about 17% difference on average for TS1 and 21% for TS2). To understand which active set do the (perturbed) Algorithm 6.1 and the (unperturbed) 3.1 predict, we terminate both algorithms at the same iteration and compare the predicted active sets with the actual optimal active sets obtained from an interior point solver and a simplex solver[12].

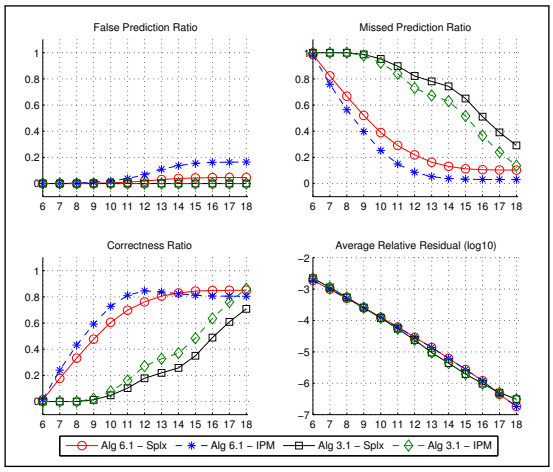



Figure 6.1: Prediction ratios for randomly generated problems

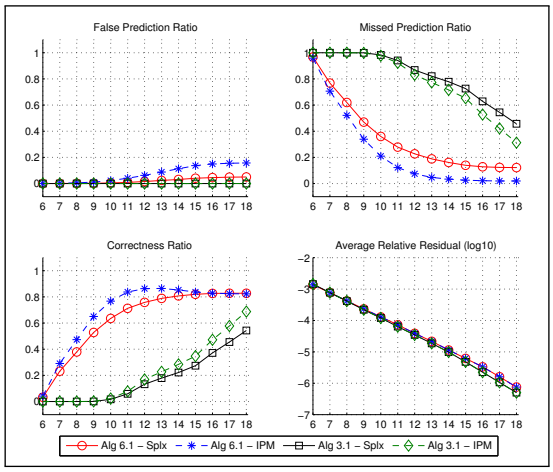



Figure 6.2: Prediction ratios for randomly generated primal-dual degenerate problems

[12] We obtain the 'actual optimal active set' by solving the problem using MATLAB's solver LINPROG with the 'algorithm' option set to interior point or simplex and considering all variables less than $10^{-5}$ as active.

In Figures 6.1 and 6.2, we present the results for TS1 (left) and TS2 (right). The x-axis shows the number of interior point iterations at which we terminate the algorithms. In each figure, the first three plots (from left to right, top to bottom) show the average value of the three measures mentioned above for the test problems in question. The last plot at the bottom right corner presents the corresponding $\log_{10}$ scaled relative KKT residuals, defined in (2.18). There are four lines in each plot, representing the prediction ratios by comparing the active set from Algorithm 6.1 with that from MATLAB's simplex solver (solid red line with circle) and from MATLAB's IPM (dashed blue line with star), and Algorithm 3.1 with simplex (solid black line with square sign) and with IPM (dashed green line with diamond sign) respectively.

- Figures 6.1 and 6.2 show that the average correctness ratios for Algorithm 6.1 are at least as good and generally better — namely, more than 4 times higher at certain iterations — than those for Algorithm 3.1. Thus it seems that using perturbations can only improve the active-set prediction capabilities of IPMs.

- Algorithm 3.1 is in fact an interior point solver applied to (PD) which approaches a strictly complementary (PD) solution. This is confirmed by having better correctness ratios when comparing Algorithm 3.1 with the IPM than when comparing it with the simplex.

- Due to the fact that the active set from the IPM (the strictly complementary partition) contains less elements than that from the simplex (vertex solution), the correctness ratio of Algorithm 6.1 compared with the IPM is higher than that compared with the simplex at the early stage. However the false-prediction ratios of the former climb up to about 0.16 at the end for both test cases. Thus the corresponding correctness ratios go down. The false-prediction ratios of comparing Algorithm 6.1 with simplex are much less, about 0.05 for both cases. The behaviour of the false-prediction ratios seems to imply that Algorithm 6.1 predicts the active set of a vertex solution (that may not be the same vertex as obtained by the simplex solver).

- After 18 iterations, the correctness ratios do not reach 1. This is due to ill-conditioning which prevents us from solving any further. For this $18^{\text{th}}$ iteration, the perturbations are not zero, they are about $\mathcal{O}(10^{-2})$ for problems in TS1 and $\mathcal{O}(10^{-3})$ for the degenerate problems in TS2, and on average the relative residual (2.18) is lower than $10^{-6}$.

**Can Algorithm 6.1 predict the optimal active set of (PD) sooner than it obtains the strictly complementary partition of ($\text{PD}_\lambda$)?** In Figures 6.3 and 6.4, besides comparing the predicted active set of (PD) with the actual active set of (PD), we also compared the predicted active set of ($\text{PD}_\lambda$)[13] with the actual active sets of ($\text{PD}_\lambda$)

[13]We apply Algorithm 3.1 to the equivalent form (4.6) of ($\text{PD}_\lambda$), which is equivalent to solving the perturbed problem using an IPM method and predicting the active set of the perturbed problem on the

obtained from a simplex solver (solid purple line with downward-pointing triangle) and an IPM solver (dashed brown line with upward-pointing triangle), respectively; see Footnote 12 on the choice of solvers. We again use the test sets TS1 and TS2.

We can see that on average Algorithm 6.1 can predict a better active set for (PD) than when applying Algorithm 3.1 to predict the active set of ($PD_\lambda$). Furthermore, for test case TS1, before iteration 12, Algorithm 3.1 cannot predict much concerning the active set of ($PD_\lambda$) while Algorithm 6.1 already has an increasingly accurate prediction for the active set of (PD) (approximately 80% of the active set of (PD) at iterations 12). We can draw similar conclusions for TS2.

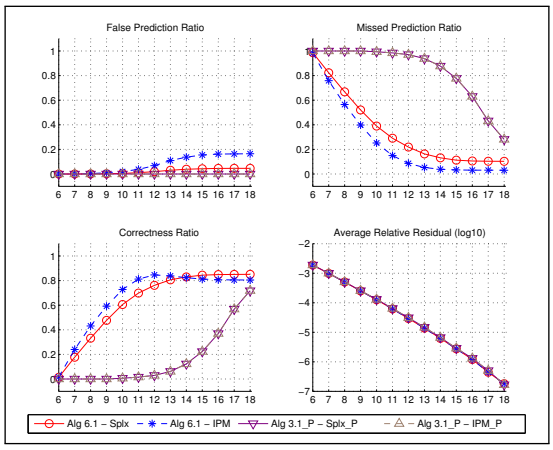



Figure 6.3: Comparing perturbed active-set predictions for TS1

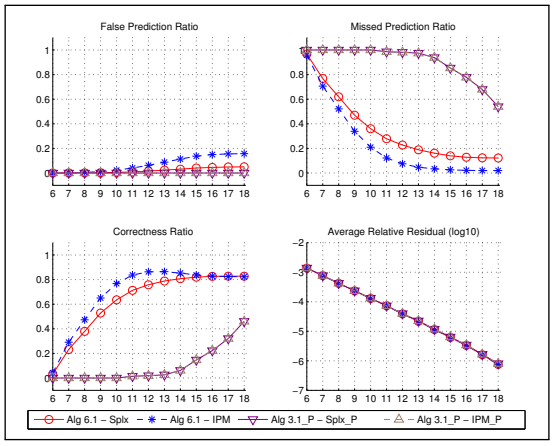



Figure 6.4: Comparing perturbed active-set predictions for TS2

Note that, to yield good performance, we do not need to force the active set of ($PD_\lambda$) to be the same as the (original) active set of (PD). In fact, for most test problems in both TS1 and TS2, this does not hold. When perturbations are not so small, namely $\mathcal{O}(10^{-2})$ or $\mathcal{O}(10^{-3})$, which is the case even in the last IPM iterations in Figures 6.3 and 6.4, the perturbed optimal active set is different from the original optimal active set for nearly all of the test problems in both TS1 and TS2; see the discussion on page 79.

Furthermore, as mentioned on page 79, almost all of the perturbed problems have a unique and nondegenerate solution, regardless of the uniqueness or degeneracy of the original test problems. This is the reason why the predictions of the perturbed active set when comparing with simplex and IPM are identical in Figures 6.3 and 6.4.

### 6.3.2 Crossover to simplex

In this section, we test the efficiency of our active-set predictions using perturbations when crossing over to a simplex solver after some IPM iterations. We choose LP_SOLVE [15] as our simplex solver. Although there are many different simplex implementations, many of which are probably more efficient and powerful than LP_SOLVE, we chose LP_SOLVE because its MATLAB interface has the functionality that allows us

way.

to set the initial basis. To the best of our knowledge, this is the only such open source simplex solver.

**Initial basis for the simplex method (Procedure 6.2).** Assume we terminate the perturbed algorithm Algorithm 6.1 at the $k^{th}$ iteration, with the predicted active set $\mathcal{A}^k$. To generate an initial basis $\mathcal{B}$ from $\mathcal{A}^k$, we first obtain all independent columns in $A_{\mathcal{I}^k}$. If this submatrix is not of rank $m$, we choose a column from $A_{\mathcal{A}^k}$ and append it to the submatrix provided it is independent of existing columns in the submatrix. The order in which columns are added back in is decided by dual information, namely we keep trying a series of columns $\{A_{i_t}\}$, where $i_t \in \mathcal{A}^k$ and $s^k_{i_1} \leq s^k_{i_2} \leq \ldots \leq s^k_{i_{|\mathcal{A}^k|}}$, until a full rank square matrix is obtained. If $A$ is full row rank[14], this procedure is finite. A similar approach has been used in [126, Section 7] to form a basis of $A$.

**Procedure 6.2** Generating an Initial Basis for the Simplex Method

**At the $k^{th}$ iteration**:
**given** predicted active and inactive sets $\mathcal{A}^k$ and $\mathcal{I}^k$, the iterate $(x^k, y^k, s^k)$ and a large constant $Const > \max(s^k)$, set $\mathcal{B} = \mathcal{I}^k$;
**if** $\text{rank}(A_\mathcal{B}) < |\mathcal{I}^k|$ **then**
  find the set of indices of all independent columns of $A_\mathcal{B}$, say $idx$.
  set $\mathcal{B} = idx$;
**end if**
**while** $\text{rank}(A_\mathcal{B}) < m$ **do**
  find the index $j = \underset{j \in \mathcal{A}^k}{\text{argmin}}\,(s^k_{\mathcal{A}^k})$ and set $(s^k_{\mathcal{A}^k})_j = Const$;
  **if** $(A_{\mathcal{A}^k})_j$ is independent with columns of $A_\mathcal{B}$ **then**
    set $\mathcal{B} = \mathcal{B} \cup \{j\}$;
  **end if**
**end while**

To conduct the tests, we first choose a threshold $\mu^{\text{cap}}_\lambda$, run Algorithm 6.1, terminate the algorithm when $\mu^k_\lambda < \mu^{\text{cap}}_\lambda$, record the number of interior point iterations, say $K$, generate an initial basis $\mathcal{B}$ by the above procedure and finally start the simplex solver LP_SOLVE from the initial basis $\mathcal{B}$. For comparison purposes we perform exactly $K$ iterations of Algorithm 3.1, and generate a new basis for (PD) by the same procedure, without constraining the value of $\mu^k$. All tests in this part are run with $\mu^{\text{cap}}_\lambda = 10^{-3}$.

We compare the number of simplex iterations used to get an optimal solution after crossover from Algorithms 6.1 and 3.1, visualising the results via a relative performance profile [103]. Namely, we consider the following relative iteration count,

$$\text{rl}_i = -\log_2 \frac{\text{Iter}^p_i}{\text{Iter}^0_i}, \tag{6.4}$$

where $i$ stands for the $i^{th}$ problem, the numerator stands for the number of simplex iterations performed after Algorithm 6.1 and the denominator measures the same but

[14] This can be guaranteed by preprocessing, as mentioned on page 40.

after Algorithm 3.1. If, for problem $i$, Algorithm 6.1 uses fewer simplex iterations, we get a positive valued bar with height $= \mathrm{rl}_i$. If Algorithm 3.1 wins, we obtain a negative valued bar with height defined as $-\mathrm{rl}_i$. The value of the bar will be 0 if these two yield the same simplex iterations or LP_SOLVE fails for both algorithms. If LP_SOLVE fails to solve problem $i$ for Algorithm 6.1, we have a negative valued bar with height of $\max_i(|\mathrm{rl}_i|)$, otherwise a positive valued bar with the same height. It is clear that the winner outperforms the loser by $2^{|\mathrm{rl}_i|}$ times and one algorithm outperforms the other by having more bars (or larger area of bars) in its direction.

**Crossover to simplex for randomly generated test problems (TS1 and TS2).** In Figures 6.5 and 6.6, we show the profiles for TS1 (left) and TS2 (right), with bars sorted from largest to smallest in height. We can see that, counting the number of simplex iterations after each algorithm, the performance of Algorithm 6.1 dominates that of Algorithm 3.1 in both cases.

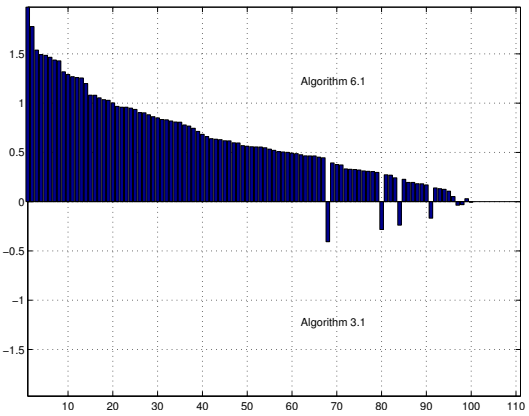



Figure 6.5: Simplex iteration count for randomly generated problems

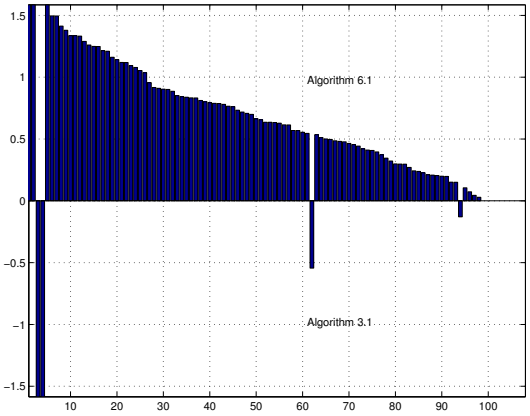



Figure 6.6: Simplex iteration count for randomly generated primal-dual degenerate problems

In Table 6.1, we show the average number of simplex iterations, the average IPM iterations and the average $\mu_\lambda^k$ and $\mu^k$ when we terminate Algorithms 6.1 and 3.1 for both test sets (TS1 and TS2). On average, using perturbations saves about 34% simplex iterations for the test case TS1 and about 37% for TS2. Due to our experimental setup, the number of IPM iterations are the same for Algorithms 6.1 and 3.1, and the average final $\mu_\lambda^k$ and $\mu^k$ before crossover are of order $10^{-4}$.[15]

We also tracked the difference between the initial bases generated from Algorithms 6.1 and 3.1. We use relative difference[16] to measure the degree of difference between two bases. On average the relative difference is over 60%, and over 90% of the

[15] The definition of $\mu_\lambda^k$ and $\mu^k$ in Algorithms 6.1 and 3.1, respectively, as well as the choice of $(x^0, s^0)$ to be identical for (PD$_\lambda$) and (PD), imply that $\mu_\lambda^0 > \mu^0$, with the difference being essentially dictated by the level of perturbations $(\lambda^0, \phi^0)$. Thus we are not making it any easier for Algorithm 6.1 compared to Algorithm 3.1 in the choice of starting point.

[16] The number of elements in either basis generated from Algorithms 6.1 or 3.1 but not both divided by the cardinality of the union of two bases.

Table 6.1: Crossover to simplex when $\mu_\lambda^k < 10^{-3}$ for random problems.

| | Primal nondegenerate (TS1) | | PD degenerate (TS2) | |
|---|---|---|---|---|
| | Algorithm 6.1 | Algorithm 3.1 | Algorithm 6.1 | Algorithm 3.1 |
| Avg simplex iterations | 287 | 436 | 292 | 464 |
| Avg IPM iterations | 10 | 10 | 10 | 10 |
| Avg $\mu_\lambda^k$ and $\mu^k$ when crossover | $7.33 \times 10^{-4}$ | $6.80 \times 10^{-4}$ | $7.53 \times 10^{-4}$ | $7.14 \times 10^{-4}$ |

test problems have greater than 50% relative difference. Thus our preliminary numerical experiments illustrate that using perturbations is likely to improve the efficiency when crossing over to simplex.

**Netlib test problems (TS3).** The good prediction performance of the perturbed algorithm is not only obtained for randomly generated problems, but also for the subset of Netlib problems (TS3). Here we add an additional termination criterion, namely we terminate both algorithms when $\mu_\lambda^k$ and $\mu^k$ are less than $10^{-3}$ or when the relative residual (2.18) less than $10^{-6}$, whichever occurs first[17].

Figure 6.7 presents the relative performance profile generated the same way as for the random tests (see (6.4) and accompanying explanation). From this figure, we can see that for over half of the test problems, Algorithm 6.1 outperforms Algorithm 3.1 by over 1.5 times. Algorithm 6.1 'loses' for only 7 problems.

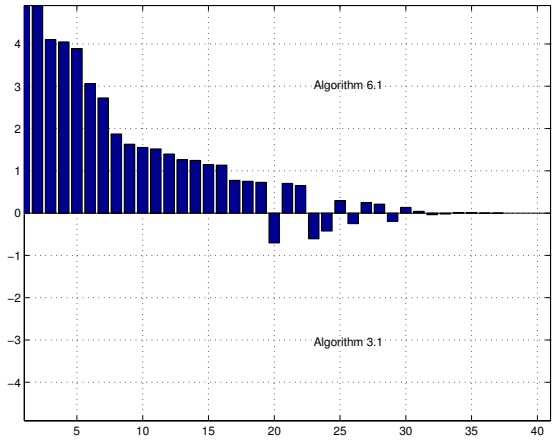



Figure 6.7: Crossover to simplex for 37 Netlib problems

We also summarise the results in Table 6.2. On average, we save about 38% simplex iterations by applying perturbations. The average numbers in the table exclude the data for SHIP08S, since LP_SOLVE fails to solve it when we do not apply perturbations.

We do not give the average value of $\mu_\lambda^k$ in the table as it is more involved than for random problems. In particular, for the problems with very large component in $b$ (problems marked by * in Table A.1 ), the value of $\mu_\lambda^k$ is greater than $10^{-3}$ for

[17] This is because some problems have very large components in the right hand side $b$ with $\max(b) > 10^3$. For these problems, even when $\mu_\lambda^k > 10^{-3}$, the relative residual may already be less than $10^{-6}$ and this causes numerical problems when trying to decrease $\mu_\lambda^k$ further. There are five problems of this kind, AGG3, FORPLAN, GROW7, ISRAEL and SHARE1B, and we have marked those problems by $*$ in Table A.1.

Table 6.2: Crossover to simplex when $\mu_\lambda^k < 10^{-3}$ for 37 Netlib problems (TS3).

| | Algorithm 6.1 | Algorithm 3.1 |
|---|---|---|
| Avg simplex iterations | 358 | 612 |
| Avg IPM iterations | 22 | 22 |

both Algorithms 6.1 and 3.1. There are 8 additional problems, including 25FV47, BNL1, BRANDY, KB2, SCFXM2, SCRS8, SCATP1 and STAIR, for which the value of $\mu_\lambda^k$ is less than $10^{-3}$ only when we apply perturbations. This seems to imply that using perturbations can somehow accelerate the interior point method procedure or yield better conditioning. Except for these particular problems, the average value of $\mu_\lambda^k$ is of order $10^{-4}$. For detailed data, see Table A.1.

As for randomly generated problems, we also tracked and compared the differences between initial bases obtained from Algorithms 6.1 and 3.1. We use the same relative difference measure (see Footnote 16). The average difference is about 40%, but there are 9 problems[18] with relative difference less than 10%. Algorithm 6.1 is no better than Algorithm 3.1 for these problems. Generally, for small problems with small relative differences between bases, the simplex iterations are quite similar; for large problems, even small relative difference can yield quite different simplex iterations (such as for SEBA and STOCFOR2). The disappointing small relative difference of initial bases may be the result of inappropriate initial perturbations, improper shrinking speed of perturbations or ill-conditioning.

## 6.4 Numerical results using the identification function for active-set prediction

In Chapter 3, we have introduced the identification function for LP and proved that it can predict the optimal active set of (PD) when the iterates are close to the solution set. A crucial outstanding question when using the identification function is that we cannot get a satisfyingly good prediction at an early stage of the IPM iterative process, which is our main motivation for developing the idea of using controlled perturbations. In Section 5.5, we have proved that the identification function also works with controlled perturbations. Here in the following two sections, we try to illustrate that using perturbations can also help the identification function predict sooner and better.

Note that apart from using the active-set prediction procedure (Procedure 3.2) with the identification function, we conduct exactly the same tests as those for cut-off (Section 6.3).

### 6.4.1 Comparing prediction ratios

In Figures 6.8 and 6.9, we present the results for prediction ratios. There are four lines in each plot, representing the prediction ratios by comparing the active set from Algo-

[18] AFIRO, AGG3, GROW7, ISREAL, SC50B, SCFXM2, SCFXM3, SEBA and STOCFOR2.

rithm 6.1 with that from MATLAB's simplex solver (solid red line with circle) and from MATLAB's IPM (dashed blue line with star), and Algorithm 3.1 with simplex (solid black line with square sign) and with IPM (dashed green line with diamond sign) respectively.

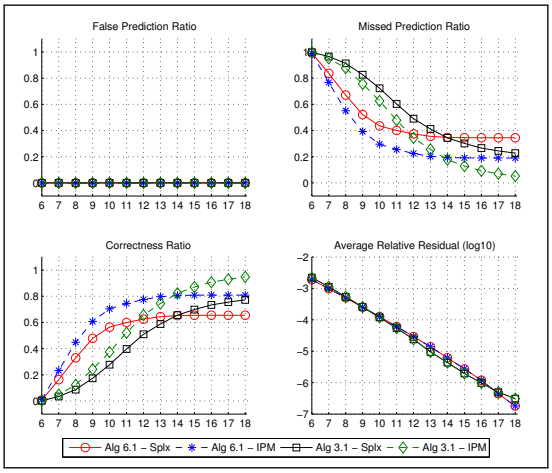



Figure 6.8: Active-set predictions for TS1 with the identification function and perturbations

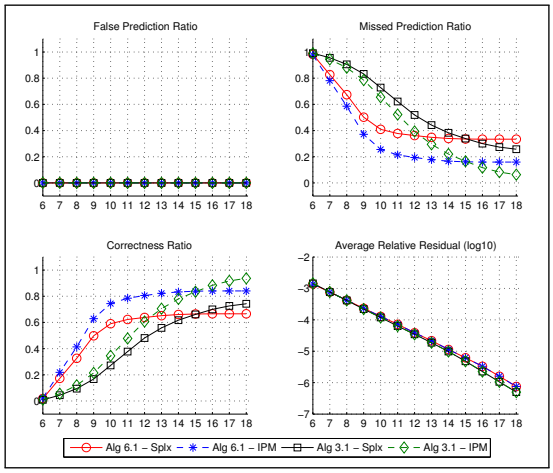



Figure 6.9: Active-set predictions for TS2 with the identification function and perturbations

- Figures 6.8 and 6.9 show that at early stages (average relative residual greater than $10^{-5}$), the average correctness ratios for Algorithm 6.1 are at least as good and generally better than those for Algorithm 3.1. Furthermore, Algorithm 6.1 with the identification function predicts almost 80% of the optimal active set at a strictly complementary solution (the strictly complementary partition) of (PD). It seems that using perturbations can improve the performance of the identification function at the early stages of the IPM process.

- The false predictions for all algorithms and test cases are zero at almost all iterations. Thus it reveals some degree of accuracy and reliability of the identification function, with or without perturbations.

- Note that the average correctness ratios of comparing Algorithm 6.1 with IPM is higher than that of comparing Algorithm 6.1 with simplex. This seems to imply that Algorithm 6.1 with the identification function predicts the active set of a strictly complementary solution. Note that in Figures 6.1 and 6.2, we observed that Algorithm 6.1 with cut-off seems to predict the active set of a vertex solution, implied by the lower average false-prediction ratios of comparing Algorithm 6.1 (with cut-off) with simplex.

- After the $14^{\text{th}}$ iteration for TS1 and the $15^{\text{th}}$ for TS2 (average relative residuals less than $10^{-5}$), the correctness ratios of comparing Algorithm 3.1 with simplex and IPM start to catch up with the corresponding ratios for Algorithm 6.1. This

is due to the perturbations not being zero even at the $18^{\text{th}}$ iteration, but staying approximately $\mathcal{O}(10^{-2})$ for problems in TS1 and $\mathcal{O}(10^{-3})$ for the degenerate problems in TS2.[19] Thus when we solve the perturbed problems further, the iterates may not get closer to the original optimal solution.

### 6.4.2 Crossover to simplex

In this section, we present the numerical results for the crossover test for combining the identification function with controlled perturbations. At the end of this section, we briefly compare the performances of using cut-off and the identification function in Procedure 3.2.

**Randomly generated test problems (TS1 and TS2).** Figures 6.10 and 6.11 illustrate that Algorithm 6.1 yields less simplex iterations than Algorithm 3.1 for the majority of the test problems.

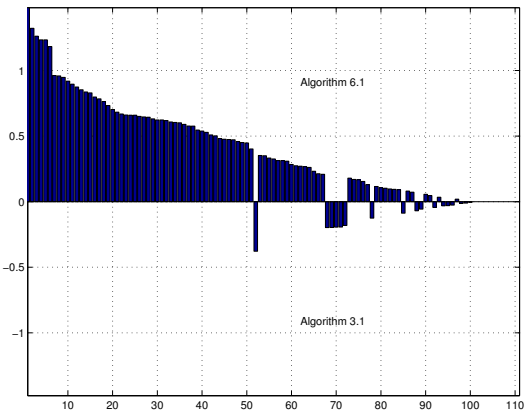



Figure 6.10: Crossover with the identification function: simplex iteration count for randomly generated problems (TS1)

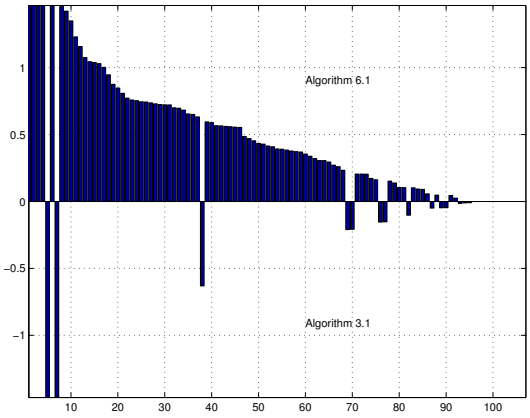



Figure 6.11: Crossover with the identification function: simplex iteration count for randomly generated primal-dual degenerate problems (TS2)

In Table 6.3, we present the average number of simplex iterations, the average IPM iterations and the average $\mu_\lambda^k$ and $\mu^k$ when we terminate Algorithms 6.1 and 3.1 with the identification function for both test sets (TS1 and TS2). On average, using perturbations saves about 25% simplex iterations for the test case TS1 and about 27% for TS2. Furthermore, on average the relative difference between the bases generated from Agorithms 6.1 and 3.1 is about 50% for both test cases. Thus using perturbations is still likely to improve the efficiency of the identification function, when crossing over to simplex.

**Netlib test problems (TS3).** Similar performances are also obtained for selected NETLIB problems; see Figure 6.12 and Table 6.4. Using perturbations saves about

[19] This phenomenon is caused by the update rules for perturbations, not ill-conditioning.

Table 6.3: Crossover to simplex for random problems (with the identification function).

| | Primal nondegenerate (TS1) | | PD degenerate (TS2) | |
|---|---|---|---|---|
| | Algorithm 6.1 | Algorithm 3.1 | Algorithm 6.1 | Algorithm 3.1 |
| Avg simplex iterations | 311 | 416 | 303 | 417 |
| Avg IPM iterations | 10 | 10 | 10 | 10 |
| Avg $\mu_\lambda^k$ and $\mu^k$ when crossover | $7.33 \times 10^{-4}$ | $6.80 \times 10^{-4}$ | $7.53 \times 10^{-4}$ | $7.14 \times 10^{-4}$ |

11% of simplex iterations. And the average relative difference of initial bases from Algorithm 6.1 and 3.1 is about 33%.

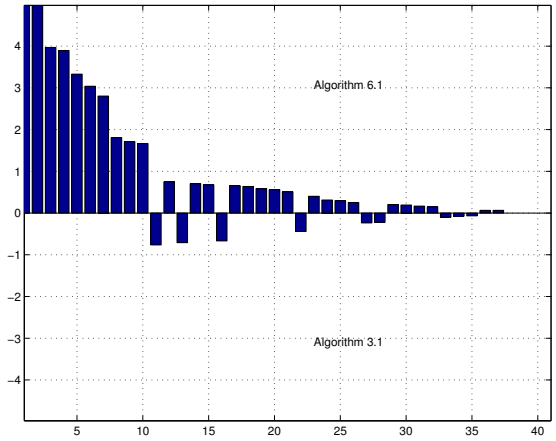



Figure 6.12: Crossover with the identification function: 37 Netlib problems

Table 6.4: Crossover to simplex for 37 Netlib problems (with the identification function).

| | Algorithm 6.1 | Algorithm 3.1 |
|---|---|---|
| Avg simplex iterations | 423 | 474 |
| Avg IPM iterations | 22 | 22 |

### 6.4.3 Comparisons between cut-off and the identification function

We have tested the use of controlled perturbations with cut-off and the identification function to predict the optimal active set of the original (PD) problem. In this section, we compare the performance of using cut-off and the identification function within the perturbed algorithm framework (Algorithm 6.1).

**Correctness ratios.** We have observed that Algorithm 6.1 with cut-off predicts the optimal active set of a vertex solution (Section 6.3.1) and that with the identification function predicts the strictly complementary partition (Section 6.4.1), while Algorithm 3.1 always predicts the strictly complementary partition. In order to conduct performance comparison, we plot corresponding correctness ratios together in Figure 6.13 for the random problems in TS1 and Figure 6.14 for the random degenerate problems in TS2, respectively. Namely we present the correctness ratios by comparing the active

set from Algorithm 6.1 using cut-off with that from MATLAB's simplex solver (solid red line with circle), Algorithm 3.1 using cut-off with MATLAB's IPM solver (dashed blue line with star), Algorithm 6.1 using the identification function with MATLAB's IPM solver (solid black line with square sign), and Algorithm 3.1 using the identification function with MATLAB's IPM solver (dashed green line with diamond sign).

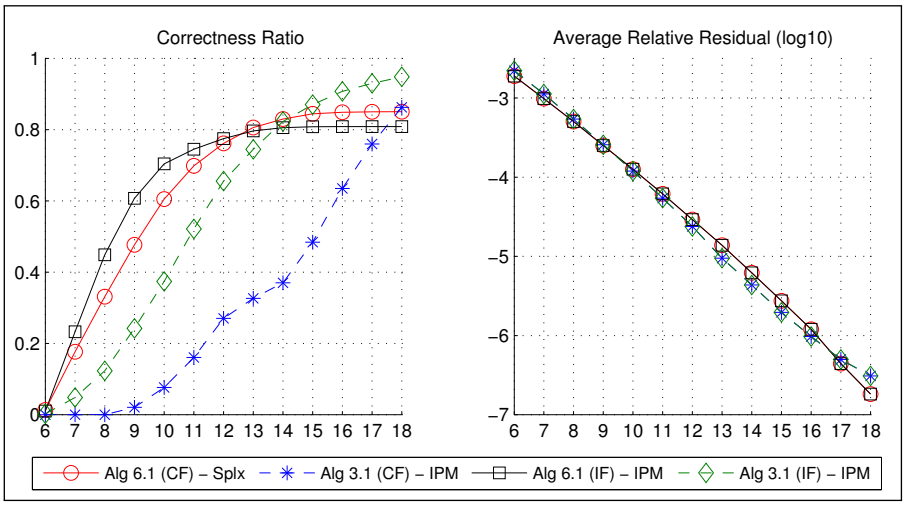



Figure 6.13: Comparing active-set prediction capabilities of cut-off and the identification function in the framework of Algorithm 6.1, for randomly generated problems in TS1

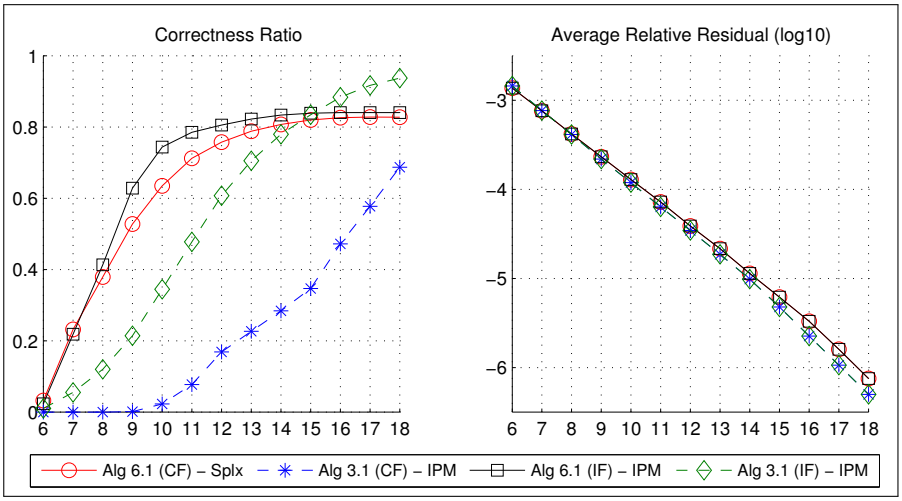



Figure 6.14: Comparing active-set prediction capabilities of cut-off and the identification function in the framework of Algorithm 6.1, for randomly generated primal-dual degenerate problems in TS2

- As we mentioned before, the algorithms with perturbations generally performs better than the algorithms without perturbations at early stages (average relative residual greater than $10^{-5}$). When the relative residual is less than $10^{-5}$,

Algorithm 3.1 with the identification function starts to outperform the others. Our aim, however, is to predict a satisfying proportion of the optimal active set of (PD) as early as possible, and thus we do not intend to solve the perturbed problems to higher accuracy than $10^{-5}$.

- It seems that using the identification function improves the accuracy of predictions for the unperturbed algorithm, Algorithm 3.1. Correctness ratios from Algorithm 3.1 with the identification function is over two times higher than that with cut-off at some iterations, for test problems in both TS1 and TS2.
- In the context of the perturbed algorithm (Algorithm 6.1), the identification function improves the accuracy of predictions as well, but the improvement is minor. Note that, to yield similar performance, using cut-off is simpler and takes slightly less computational effort than using the identification function.

**Crossover to simplex.** By comparing corresponding iteration data in Table 6.1 with that in Table 6.3 and data in Table 6.2 with Table 6.4, we observe the following.

- **The average number of simplex iterations from (the unperturbed) Algorithm 3.1 with the identification function is less than this algorithm with cut-off**, 5% less for problems in TS1, 10% for TS2 and 23% for TS3. This is due to Algorithm 3.1 with cut-off and with the identification function both predicting the strictly complementary partition of (PD) and the latter yields better correctness ratios (see the dashed lines in Figures 6.13 and 6.14). The better the predictions are, the better the initial basis could be, when they predict the same active set.
- However, the situation is different when using perturbations. **On average Algorithm 6.1 with the identification function causes more simplex iterations than that with cut-off**, 8% more for problems in TS1, 4% for TS2 and 18% for TS3. A possible explanation is that the former predicts the strictly complementary partition of (PD) while the latter predicts the active set at a vertex solution of (PD). As the simplex method starts from vertices, the initial basis generated from the former may be worse and less suitable than that from the latter, despite the average correctness ratios of the former being better than that of the latter (see the solid lines in Figures 6.13 and 6.14).

## 6.5 Conclusions

In this chapter, we have provided preliminary numerical evidence that our approach to active-set prediction for IPMs looks promising in that the perturbed problems are not being solved to high accuracy before the original optimal active set can be accurately predicted and that the perturbations help with the accuracy and speed of the activity

prediction for the original solution set. Notably, the perturbed algorithm managed to predict on average over 70% of the original optimal active set when the average relative residuals reach $10^{-4}$. Note that as we are applying a standard IPM to a suite of perturbed problems, our approach maintains polynomial complexity provided the number of times we shrink the perturbations is (polynomially) finite.

In Section 5.3.2, we have presented the situations when our approach allows an earlier prediction of the original active set as compared to the case when we solve and predict the original LP directly (Theorem 5.13 in Section 5.3). However these results reply on some conditions which are not imposed in our numerical tests. In order to better understand why the perturbed algorithm is generally better than the unperturbed one in the numerical tests, we also have the following conjectures.

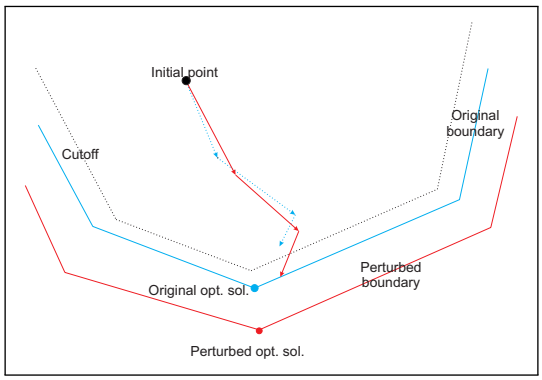



Figure 6.15: Illustration for potential larger Newton steps for the perturbed algorithm. The blue dashed line stands for the iterates of the unperturbed algorithm and the red solid line for the perturbed.

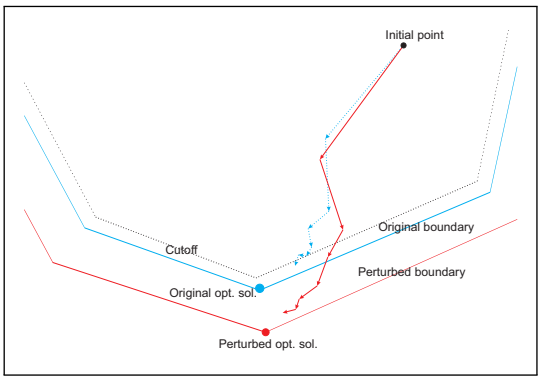



Figure 6.16: Illustration for possible behaviours of the iterates when they are close to the threshold and the boundaries. Blue dashed line for the unperturbed algorithm and red solid line for the perturbed as well.

- By enlarging the feasible set using perturbations, we potentially allow larger Newton steps during the first few iterations, when the perturbations are not very small. Therefore the iterates from the perturbed algorithm stand better chance to move across the threshold sooner, recalling that the perturbed and unperturbed algorithms are using the same cutoff. See Figure 6.15 for an illustration.

- From the study of the indicators [32], we know when an IPM iterate, say $x_i^k$, approaches its boundary, the corresponding Newton direction $\Delta x_i^k$ may only make small improvement towards the original optimal solution and then may suffer from difficulty in moving below the threshold; when an iterate of the perturbed algorithm approaches the threshold, the iterate could be still far from the boundary of the perturbed problem and therefore it can pass the threshold without 'slowing down'. See Figure 6.16 for an illustration.

# 7

# Active-set Prediction for Quadratic Programming Problems

In this chapter, we extend the idea of using controlled perturbations to convex Quadratic Programming (QP) problems. QP problems share many properties of LP, based on which the extension of some results is straightforward. However, QP problems are not guaranteed to have a strictly complementary solution and the existence of a strictly complementary solution is crucial to the theory for the LP case. In the proof of Theorem 4.7, the construction of an optimal solution of ($PD_\lambda$) relies on the existence of a strictly complementary (PD) solution, more exactly the strictly complementary partition for the solution of (PD); without this, Lemma 5.2 will not hold and therefore the consequent Lemmas 5.3 and 5.5 and the main prediction results, Theorems 5.6 – 5.8, will not hold.

We start this chapter by presenting the formulations of the perturbed QP problems (Section 7.1) and their properties (Section 7.2). We then derive theorems on predicting the optimal active set of a QP problem without the strictly complementary assumption (Section 7.3.1); we also present results on predicting the optimal tripartition of a QP problem (Section 7.3.2) and prediction results using the identification function (Section 7.3.3). In Section 7.4, we first present the perturbed algorithm structure in Section 7.4.1 and introduce the test problems in Section 7.4.2. In Section 7.4.3, similarly to the linear case, we conduct numerical tests on the accuracy of the predicted optimal active set. Then in Section 7.4.4, we predict the optimal active set, build a sub-problem by removing the active constraints and corresponding rows/columns in the problem data, $A$, $c$, and $H$, solve the sub-problem using the active-set method and compare the number of active-set iterations. The relative difference between the optimal objective value of the sub-problem and that of the original problem is also measured. In Section 7.4.5, we compare the performance of the perturbed algorithm using cut-off with that using the identification function.

## 7.1 Perturbed quadratic programming problems

Because of the similarity to LP, we only briefly describe the perturbed problems and corresponding terms and properties for QP. We enlarge the feasible set of the (QPD)

problems (see page 26) by enlarging the nonnegativity constraints in (QPD) and consider the following perturbed problems,

$$\begin{array}{lll} \text{(Primal)} & & \\ \min_x & \frac{1}{2}(x+\lambda)^T H(x+\lambda) & \\ & +(c+(I-H)\lambda)^T(x+\lambda) & \\ \text{s.t.} & Ax = b, & \\ & x \geq -\lambda, & \end{array} \qquad \begin{array}{lll} \text{(Dual)} & & \\ \max_{(x,y,s)} & (b+A\lambda)^T y & \\ & -\frac{1}{2}(x+\lambda)^T H(x+\lambda) & \\ \text{s.t.} & A^T y + s - Hx = c, & \\ & y \text{ free}, \ s \geq -\lambda, & \end{array} \tag{QPD$_\lambda$}$$

where $\lambda \in \mathbb{R}^n$ and $\lambda \geq 0$. Note that if $\lambda \equiv 0$, (QPD$_\lambda$) is equivalent to (QPD). By formulating the Lagrangian dual [17] of the primal (dual) problem in (QPD$_\lambda$), it is straightforward to show that the two problems in (QPD$_\lambda$) are dual to each other.

We denote the set of *strictly feasible points* of (QPD$_\lambda$)

$$\mathcal{QF}_\lambda^0 = \{(x,y,s) \,|\, Ax = b,\ A^T y + s - Hx = c,\ x+\lambda > 0,\ s+\lambda > 0\}. \tag{7.1}$$

$\mathcal{QF}_\lambda^0$ coincides with the strictly feasible set (2.26) of (QPD) if $\lambda \equiv 0$.

According to [105, Theorem 12.1], we derive the KKT conditions for (QPD$_\lambda$),

$$Ax = b, \tag{7.2a}$$
$$A^T y + s - Hx = c, \tag{7.2b}$$
$$(X+\Lambda)(S+\Lambda)e = 0, \tag{7.2c}$$
$$(x+\lambda, s+\lambda) \geq 0, \tag{7.2d}$$

where $\Lambda$ is a diagonal matrix with the entries of $\lambda$ on the diagonal and $e$ is a vector of ones. Any primal-dual pair $(x,y,s)$ is an optimal solution of (QPD$_\lambda$) if and only if it satisfies (7.2).

**Equivalent formulation of (QPD$_\lambda$).** Let $p = x + \lambda$ and $q = s + \lambda$. Then we can rewrite (QPD$_\lambda$) as follows,

$$\begin{array}{lll} \text{(Primal)} & & \\ \min_p & \frac{1}{2}p^T H p + \hat{c}_\lambda^T p & \\ \text{s.t.} & Ap = \hat{b}_\lambda, & \\ & p \geq 0, & \end{array} \qquad \begin{array}{lll} \text{(Dual)} & & \\ \max_{(p,y,q)} & \hat{b}_\lambda^T y - \frac{1}{2}p^T H p & \\ \text{s.t.} & A^T y + q - Hp = \hat{c}_\lambda, & \\ & y \text{ free}, \ q \geq 0, & \end{array} \tag{7.3}$$

where

$$\hat{b}_\lambda = b + A\lambda \quad \text{and} \quad \hat{c}_\lambda = c + (I-H)\lambda. \tag{7.4}$$

Formulating the KKT conditions of (7.3) and comparing them with (7.2), we have the following result.

**Proposition 7.1.** *$(x_\lambda^*, y_\lambda^*, s_\lambda^*)$ is an optimal solution of (QPD$_\lambda$) with some $\lambda \geq 0$ if and only if $(p_\lambda^*, y_\lambda^*, q_\lambda^*)$, with $p_\lambda^* = x_\lambda^* + \lambda$ and $q_\lambda^* = s_\lambda^* + \lambda$, is a solution of (7.3).*

**The central path of** (QPD$_\lambda$)**.** Following [21, Chapter 11], we derive the central path equations for (QPD$_\lambda$), namely

$$\begin{aligned} Ax &= b, \\ A^T y + s - Hx &= c, \\ (X+\Lambda)(S+\Lambda)e &= \mu\, e, \\ (x+\lambda, s+\lambda) &> 0, \end{aligned} \tag{7.5}$$

where $\mu > 0$ is the barrier parameter for (QPD$_\lambda$). In [101], Monteiro and Adler show that the central path of a QP problem exists if its strictly feasible set is nonempty. From this statement and considering the equivalent form (7.3) of (QPD$_\lambda$), it follows that the central path of (QPD$_\lambda$) exists if its strictly feasible set $\mathcal{QF}_\lambda^0$ in (7.1) is nonempty. Thus we can draw the same conclusion as in the LP case, that given $\lambda > 0$, the existence of the perturbed central path requires weaker assumptions compared to those for the central path of (QPD), because $\mathcal{QF}_\lambda^0$ is nonempty if (QPD) has a nonempty primal-dual feasible set.

## 7.2 Properties of the perturbed quadratic programming problems

### 7.2.1 Perfect and relaxed perturbations

For the LP case, we know that the optimal solution of the original problems can lie on or near the central path of the perturbed problems (Section 4.2.1 or [25, Section 3.1]). Following exactly the same approach, we can verify that these results also hold for QP.

**Theorem 7.2 (Existence of 'perfect' perturbations for QP).** *Assume* (2.9) *holds and* $(x^*, y^*, s^*)$ *is a solution of* (QPD)*. Let* $\hat{\mu} > 0$*. Then there exist perturbations*

$$\hat{\lambda} = \hat{\lambda}(x^*, s^*, \hat{\mu}) > 0,$$

*such that the perturbed central path* (7.5) *with* $\lambda = \hat{\lambda}$ *passes through* $(x^*, y^*, s^*)$ *exactly when* $\mu = \hat{\mu}$*.*

**Theorem 7.3 (Existence of relaxed perturbations for QP).** *Assume* $(x^*, y^*, s^*)$ *is a solution of* (QPD)*. Let* $\hat{\mu} > 0$ *and* $\xi \in (0, 1)$*. Then there exist constants* $\hat{\lambda}_L = \hat{\lambda}_L(x^*, s^*, \hat{\mu}, \xi) > 0$*, and* $\hat{\lambda}_U = \hat{\lambda}_U(x^*, s^*, \hat{\mu}, \xi) > 0$*, such that*

*for $\hat{\lambda}_L \leq \lambda \leq \hat{\lambda}_U$, $(x^*, y^*, s^*)$ is strictly feasible for (QPD) and satisfies*

$$\xi\hat{\mu}e \leq (X^* + \Lambda)(S^* + \Lambda)e \leq \frac{1}{\xi}\hat{\mu}e.$$

Intuitively, these existence theorems imply that when the perturbations are chosen properly, the perturbed central path may pass or get very close to the original optimal solution. Thus we have the hope that from the iterates which follow the perturbed central path, we may be able to get enough information about the original optimal solution, so as to predict the optimal active set of the original problem.

### 7.2.2 Preserving the optimal active sets and tripartitions

**Preserving the optimal active sets.** Recall that in (2.28) we have defined for (QPD) the primal active set $\mathcal{A}(x^*)$, the primal inactive set $\Theta(x^*)$, the dual active set $\mathcal{I}(s^*)$ and the dual inactive set $\mathcal{A}^+(s^*)$. Similarly, given a primal-dual pair $(x, y, s)$ for ($\text{QPD}_\lambda$), we define the following sets

$$\begin{aligned} \mathcal{A}_\lambda(x) &= \{i \in \{1, \ldots, n\} \,|\, x_i = -\lambda\}, \quad \Theta_\lambda(x) = \{i \in \{1, \ldots, n\} \,|\, x_i > -\lambda\}, \\ \mathcal{I}_\lambda(s) &= \{i \in \{1, \ldots, n\} \,|\, s_i = -\lambda\}, \quad \mathcal{A}^+_\lambda(s) = \{i \in \{1, \ldots, n\} \,|\, s_i > -\lambda\}. \end{aligned} \tag{7.6}$$

In the following theorem, we show that there exists a primal-dual pair of points which is close to the optimal solution of ($\text{QPD}_\lambda$) and the corresponding active and inactive sets at this point are the same as the optimal active and inactive sets at an optimal solution of (QPD).

**Theorem 7.4.** *Assume $(x^*, y^*, s^*)$ is an optimal solution of (QPD). Then there exist positive constants $\hat{\lambda} = \hat{\lambda}(H, A, b, c, x^*, s^*)$ and $C_3 = C_3(H, A, x^*, s^*)$, depending on the original problem (QPD), such that for every $\lambda$ where $0 < \|\lambda\| < \hat{\lambda}$, there exists a primal-dual pair $(x, y, s)$ which satisfies (7.2c, 7.2d) and we have*

$$\mathcal{A}_\lambda(x) = \mathcal{A}(x^*), \quad \Theta_\lambda(x) = \Theta(x^*), \quad \mathcal{I}_\lambda(s) = \mathcal{I}(s^*), \quad \mathcal{A}^+_\lambda(s) = \mathcal{A}^+(s^*), \tag{7.7}$$

*and*

$$\max\left(\|Ax - b\|, \|A^T y + s - Hx - c\|\right) < C_3\|\lambda\|. \tag{7.8}$$

**Proof.** We work with the equivalent form (7.3) of the problems in ($\text{QPD}_\lambda$). For convenience, for the rest of this proof, we neglect the dependency of the index sets on $(x^*, y^*, s^*)$ and use $\mathcal{A}$, $\Theta$, $\mathcal{I}$ and $\mathcal{A}^+$ to denote the partition of a matrix or a vector in accordance with the corresponding sets. Since $(x^*, y^*, s^*)$ is a solution

of (QPD) and from (2.28), we have

$$\begin{aligned} x^*_{\mathcal{A}} = 0, \quad x^*_{\Theta} > 0 \quad \text{and} \quad s^*_{\mathcal{I}} = 0, \quad s^*_{\mathcal{A}^+} > 0, \\ A_{\Theta} x^*_{\Theta} = b, \quad A^T_{\mathcal{I}} y^* - H_{\mathcal{I}\Theta} x^*_{\Theta} = c_{\mathcal{I}}, \quad A^T_{\mathcal{A}^+} y^* + s^*_{\mathcal{A}^+} - H_{\mathcal{A}^+\Theta} x^*_{\Theta} = c_{\mathcal{A}^+}, \end{aligned} \tag{7.9}$$

where $H_{XY}$ denotes $(H_{ij})_{i\in X, j\in Y}$. We define a point $(\hat{p}, \hat{y}, \hat{q})$ to be

$$\begin{aligned} \hat{p}_{\mathcal{A}} = 0, \quad \hat{p}_{\Theta} = x^*_{\Theta} + \lambda_{\Theta} + \hat{u}, \\ \hat{y} = y^* + \hat{v}, \quad \hat{q}_{\mathcal{I}} = 0, \quad \hat{q}_{\mathcal{A}^+} = s^*_{\mathcal{A}^+} + \lambda_{\mathcal{A}^+} - H_{\mathcal{A}^+\mathcal{A}} \lambda_{\mathcal{A}} - A^T_{\mathcal{A}^+} \hat{v} + H_{\mathcal{A}^+\Theta} \hat{u}, \end{aligned} \tag{7.10}$$

where $(\hat{u}, \hat{v})$ is the minimal least-squares solution of

$$M \begin{bmatrix} u \\ v \end{bmatrix} = W \begin{bmatrix} \lambda_{\mathcal{A}} \\ \lambda_{\mathcal{I}} \end{bmatrix}, \text{ with } M = \begin{bmatrix} A_{\Theta} & 0 \\ -H_{\mathcal{I}\Theta} & A^T_{\mathcal{I}} \end{bmatrix} \text{ and } W = \begin{bmatrix} A_{\mathcal{A}} & 0 \\ -H_{\mathcal{I}\mathcal{A}} & I_{\mathcal{I}} \end{bmatrix}. \tag{7.11}$$

We are about to find conditions on $\lambda$ under which $\hat{p}_{\Theta} > 0$ and $\hat{q}_{\mathcal{A}^+} > 0$, and thus we can have (7.2c), (7.2d) and (7.7) hold. From [23, Theorem 2.2.1], we have

$$\begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} = M^+ W \begin{bmatrix} \lambda_{\mathcal{A}} \\ \lambda_{\mathcal{I}} \end{bmatrix},$$

where $M^+$ is the pseudo-inverse of $M$. This and norm properties give us

$$\|(\hat{u}, \hat{v})\| \leq \|M^+ W\| \cdot (\|\lambda_{\mathcal{A}}\| + \|\lambda_{\mathcal{I}}\|) \leq 2\|M^+ W\| \cdot \|\lambda\|. \tag{7.12}$$

Let

$$\hat{\lambda} = \min \left( \frac{\min \left[ x^*_{\Theta} \; s^*_{\mathcal{A}^+} \right]}{2\|M^+ W\|}, \frac{\min \left[ x^*_{\Theta} \; s^*_{\mathcal{A}^+} \right]}{\|H_{\mathcal{A}^+\mathcal{A}}\| + 2 \left( \|A^T_{\mathcal{A}^+}\| + \|H_{\mathcal{A}^+\Theta}\| \right) \|M^+ W\|} \right),$$

where $\min \left[ x^*_{\Theta} \; s^*_{\mathcal{A}^+} \right]$ denotes the smallest elements of the vectors $x^*_{\Theta}$ and $s^*_{\mathcal{A}^+}$. This, (7.10), (7.12), $0 < \|\lambda\| < \hat{\lambda}$ and norm properties give us that

$$\hat{p}_{\Theta} \geq x^*_{\Theta} + \hat{u} \geq x^*_{\Theta} - \|\hat{u}\| e_{\Theta} > x^*_{\Theta} - 2\hat{\lambda} \|M^+ W\| e_{\Theta} \geq 0$$

and

$$\begin{aligned} \hat{q}_{\mathcal{A}^+} &\geq s^*_{\mathcal{A}^+} - H_{\mathcal{A}^+\mathcal{A}} \lambda_{\mathcal{A}} - A^T_{\mathcal{A}^+} \hat{v} + H_{\mathcal{A}^+\Theta} \hat{u} \\ &\geq s^*_{\mathcal{A}^+} - (\|H_{\mathcal{A}^+\mathcal{A}}\| \cdot \|\lambda\| + \|A^T_{\mathcal{A}^+}\| \cdot \|\hat{v}\| + \|H_{\mathcal{A}^+\Theta}\| \cdot \|\hat{u}\|) \\ &> s^*_{\mathcal{A}^+} - \left( \|H_{\mathcal{A}^+\mathcal{A}}\| + 2\|M^+ W\| (\|A^T_{\mathcal{A}^+}\| + \|H_{\mathcal{A}^+\Theta}\|) \right) \hat{\lambda} \geq 0. \end{aligned}$$

It remains to prove (7.8). From (7.4), (7.9), and (7.10), we can verify

$$\begin{aligned} A\hat{p} - \hat{b}_\lambda &= A_\Theta \hat{u} - A_\mathcal{A}\lambda_\mathcal{A} \\ &= \left( M_1 \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} - W_1 \begin{bmatrix} \lambda_\mathcal{A} \\ \lambda_\mathcal{I} \end{bmatrix} \right), \\ A_\mathcal{I}^T \hat{y} + \hat{q}_\mathcal{I} - H_{\mathcal{I}\Theta}\hat{p}_\Theta - (\hat{c}_\lambda)_\mathcal{I} &= -H_{\mathcal{I}\Theta}\hat{u} + A_\mathcal{I}^T \hat{v} + H_{\mathcal{I}\mathcal{A}}\lambda_\mathcal{A} - \lambda_\mathcal{I} \\ &= \left( M_2 \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} - W_2 \begin{bmatrix} \lambda_\mathcal{A} \\ \lambda_\mathcal{I} \end{bmatrix} \right), \\ A_{\mathcal{A}^+}^T \hat{y} + \hat{q}_{\mathcal{A}^+} - H_{\mathcal{A}^+\Theta}\hat{p}_\Theta - (\hat{c}_\lambda)_{\mathcal{A}^+} &= 0, \end{aligned} \tag{7.13}$$

where

$$M_1 = \begin{bmatrix} A_\Theta & 0 \end{bmatrix}, M_2 = \begin{bmatrix} -H_{\mathcal{I}\Theta} & A_\mathcal{I}^T \end{bmatrix}, W_1 = \begin{bmatrix} A_\mathcal{A} & 0 \end{bmatrix} \quad \text{and} \quad W_2 = \begin{bmatrix} -H_{\mathcal{I}\mathcal{A}} & I_\mathcal{I} \end{bmatrix}.$$

Since $(\hat{u}, \hat{v})$ is the least-squares solution of (7.11),

$$M = \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} \quad \text{and} \quad W = \begin{bmatrix} W_1 \\ W_2 \end{bmatrix},$$

we have

$$\begin{aligned} \|A\hat{p} - \hat{b}_\lambda\| &\le \left\| M \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} - W \begin{bmatrix} \lambda_\mathcal{A} \\ \lambda_\mathcal{I} \end{bmatrix} \right\| \le \left\| W \begin{bmatrix} \lambda_\mathcal{A} \\ \lambda_\mathcal{I} \end{bmatrix} \right\| \le 2\|W\|\|\lambda\|, \\ \|A_\mathcal{I}^T \hat{y} + \hat{q}_\mathcal{I} - H_{\mathcal{I}\Theta}\hat{p}_\Theta - (\hat{c}_\lambda)_\mathcal{I}\| &\le \left\| M \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} - W \begin{bmatrix} \lambda_\mathcal{A} \\ \lambda_\mathcal{I} \end{bmatrix} \right\| \le \left\| W \begin{bmatrix} \lambda_\mathcal{A} \\ \lambda_\mathcal{I} \end{bmatrix} \right\| \le 2\|W\|\|\lambda\|. \end{aligned}$$

This and (7.13) imply that

$$\max\left( \|A\hat{p} - \hat{b}_\lambda\|, \|A^T\hat{y} + \hat{q} - H\hat{p} - \hat{c}_\lambda\| \right) \le 2\|W\|\|\lambda\|. \tag{7.14}$$

□

**Remarks on Theorem 7.4.**

- The point $(x, y, s)$ satisfies the bound (7.2d) on $(x, s)$ and the complementary condition (7.2c). Thus the error (7.8) in the equality constraints (7.2a, 7.2b) also bounds the 'distance' between $(x, y, s)$ and the optimal solution set of (QPD$_\lambda$). This feasibility error (7.8) goes to 0 as $\lambda \to 0$, and so primal and dual feasibility can be approximately achieved. Note that, the feasibility error comes from the residual of the least problem (7.11), in other words, if (7.11) has a solution, $(x, y, s)$ will be an optimal solution of (QPD$_\lambda$) with $\lambda > 0$, at which the primal-dual active sets of (QPD$_\lambda$) are the same as the original (QPD).

- (7.14) gives an upper bound on the feasibility constraints of the equivalent form (7.3) of ($\text{QPD}_\lambda$). Setting $\hat{x} = \hat{p} - \lambda$ and $\hat{s} = \hat{q} - \lambda$, we can see this bound is also an upper bound for the feasibility constraints of (QPD).

**Preserving the optimal tripartition.** In (2.30), we have defined the complement of the optimal primal and dual inactive sets. Similarly, we denote

$$\mathcal{T}_\lambda(x,s) = \{1,\dots,n\} \setminus \left(\mathcal{A}_\lambda^+(s) \cup \Theta_\lambda(x)\right), \tag{7.15}$$

where $\mathcal{A}_\lambda^+(s)$ and $\Theta_\lambda(x)$ are defined in (7.6). Note that without the complementary condition, $(\mathcal{A}_\lambda^+(s), \Theta_\lambda(x), \mathcal{T}_\lambda(x,s))$ may not form a tripartition of the full index set. In the following corollary, we show that under certain conditions on the perturbations, there exists a primal-dual pair which is close to (ultimately in) the solution set of ($\text{QPD}_\lambda$), such that $(\mathcal{A}_\lambda^+(s), \Theta_\lambda(x), \mathcal{T}_\lambda(x,s))$ forms a tripartition and it is the same as the tripartition $(\mathcal{A}^+(s^*), \Theta(x^*), \mathcal{T}(x^*,s^*))$ at an optimal solution $(x^*,y^*,s^*)$ of (QPD).

**Corollary 7.5.** *Assume $(x^*,y^*,s^*)$ is an optimal solution of (QPD). Then there exist a positive constant $\hat{\lambda} = \hat{\lambda}(H,A,b,c,x^*,s^*)$, a positive constant $C_3 = C_3(H,A,x^*,s^*)$ and a primal-dual pair $(x,y,s)$ which satisfies (7.2c, 7.2d) with $0 < \|\lambda\| < \hat{\lambda}$, such that $(\mathcal{A}_\lambda^+(s), \Theta_\lambda(x), \mathcal{T}_\lambda(x,s))$ forms a tripartition of $\{1,\dots,n\}$ and is the same as the partition $(\mathcal{A}^+(s^*), \Theta(x^*), \mathcal{T}(x^*,s^*))$ for the original (QPD) with (7.8) satisfied, where $\mathcal{A}^+(s^*)$ and $\Theta(x^*)$ are defined in (2.28), $\mathcal{T}(x^*,s^*)$ in (2.30), $\mathcal{A}_\lambda^+(s)$ and $\Theta_\lambda(x)$ in (7.6) and $\mathcal{T}_\lambda(x,s)$ in (7.15).*

**Proof.** Recalling the definitions of $\mathcal{T}(x^*,s^*)$ and $\mathcal{T}_\lambda(x,s)$, the results follow from Theorem 7.4. □

Corollary 7.5 shows that under the same conditions for Theorem 7.4, there exists a point that is close to the solution set of the perturbed problems and preserves the optimal tripartition of the original QP. This point may be an optimal solution of ($\text{QPD}_\lambda$) as well.

## 7.3 Active-set prediction for (QPD) using perturbations

We have derived an error bound for QP in Theorem 2.14 and the following lemma simplifies that result by using feasibility conditions.

**Lemma 7.6 (Error bound for (QPD)).** *Let $(x,y,s) \in \mathcal{QF}_\lambda^0$, where $\mathcal{QF}_\lambda^0$ is defined in (7.1), and $\lambda \geq 0$. Then there exists an optimal solution $(x^*,y^*,s^*)$ of (QPD) such that*

$$\|x - x^*\| \leq \tau_p(r(x,s) + w(x,s)) \quad \text{and} \quad \|s - s^*\| \leq \tau_d(r(x,s) + w(x,s)), \tag{7.16}$$

*where $\tau_p$ and $\tau_q$ are problem-dependent constants independent of $(x,y,s)$ and $(x^*,y^*,s^*)$,*

*and*

$$r(x,s) = \| \min\{x,s\} \| \quad \textit{and} \quad w(x,s) = \|(-x,-s,x^T s)_+\|, \tag{7.17}$$

*and where* $\min\{x,s\} = (\min(x_i,s_i))_{i=1,\dots,n}$ *and* $(x)_+ = (\max(x_i,0))_{i=1,\dots,n}$.

**Proof.** Considering $Ax = b$ and $A^T y + s - Hx = c$, this follows from Theorem 2.14. □

We define a symmetric neighbourhood [54] of the perturbed central path (7.5),

$$\mathcal{N}(\gamma,\lambda) = \left\{ (x,y,s) \in \mathcal{QF}^0_\lambda \,|\, \gamma\mu_\lambda \le (x_i+\lambda_i)(s_i+\lambda_i) \le \frac{\mu_\lambda}{\gamma},\ i = 1,\dots,n \right\}, \tag{7.18}$$

where $\gamma \in (0,1)$ and $\mu_\lambda$ is defined in (5.4). In the following analysis of predicting the optimal active set (Section 7.3.1) and tripartition (Section 7.3.2), we always consider points in this neighbourhood.

**Lemma 7.7.** *Let* $(x,y,s) \in \mathcal{N}(\gamma,\lambda)$ (7.18) *for some* $\lambda \ge 0$ *and* $\mu_\lambda$ *defined in* (5.4). *Then there exists a solution* $(x^*,y^*,s^*)$ *of* (QPD) *and problem-dependent constants* $\tau_p$ *and* $\tau_d$ *that are independent of* $(x,y,s)$ *and* $(x^*,y^*,s^*)$, *such that*

$$\begin{aligned} \|x - x^*\| &< \tau_p \left(C_4\sqrt{\mu_\lambda}\max(\sqrt{\mu_\lambda},1) + 4\|\lambda\|\max(\|\lambda\|,1)\right), \\ \|s - s^*\| &< \tau_d \left(C_4\sqrt{\mu_\lambda}\max(\sqrt{\mu_\lambda},1) + 4\|\lambda\|\max(\|\lambda\|,1)\right), \end{aligned} \tag{7.19}$$

*where*

$$C_4 = \sqrt{\frac{n}{\gamma}} + n. \tag{7.20}$$

**Proof.** Since $w(x,s)$ in (7.17) has the same form as that in (5.1), then following the same proof of Lemma 5.4, we have

$$w(x,s) \le n\mu_\lambda + 2\|\lambda\| + \|\lambda\|^2. \tag{7.21}$$

It remains to find an upper bound for $r(x,s)$ in (7.17). Since $(x_i+\lambda_i)(s_i+\lambda_i) \le \frac{1}{\gamma}\mu_\lambda$, if $x_i + \lambda_i \le s_i + \lambda_i$, we have

$$0 < x_i + \lambda_i \le \frac{\mu_\lambda}{\gamma(s_i+\lambda_i)} \le \frac{\mu_\lambda}{\gamma(x_i+\lambda_i)},$$

namely $0 < x_i + \lambda_i \le \sqrt{\frac{\mu_\lambda}{\gamma}}$. Similarly if $x_i + \lambda_i > s_i + \lambda_i$, we also have $0 < s_i + \lambda_i < \sqrt{\frac{\mu_\lambda}{\gamma}}$. Thus $0 < \min\{x+\lambda, s+\lambda\} \le \sqrt{\frac{\mu_\lambda}{\gamma}}e$. So from (7.17) we have

$$r(x,s) = \| \min\{x+\lambda, s+\lambda\} - \lambda\| \le \| \min\{x+\lambda, s+\lambda\} \| + \|\lambda\| \le \sqrt{\tfrac{n\mu_\lambda}{\gamma}} + \|\lambda\|. \tag{7.22}$$

The bounds in (7.19) follow from (7.16), (7.21), and (7.22). □

### 7.3.1 Predicting the original optimal active set

Recall $\bar{\mathcal{A}}(x)$ and $\bar{\mathcal{A}}^+(s)$ defined in (5.12) for (PD). We employ the same notation for (QPD$_\lambda$) and consider $\bar{\mathcal{A}}(x)$ as the predicted active set of the original problem (QPD) and $\bar{\mathcal{A}}^+(s)$ as the predicted strongly active set of (QPD) at the primal-dual pair $(x, y, s)$. We show that prediction results for LP (Theorems 5.6 – 5.8 in Chapter 5) can be extended to the QP case, namely, under certain conditions, the active sets $\mathcal{A}(x^*)$ and $\mathcal{A}^+(s^*)$ at some solution $(x^*, y^*, s^*)$ of (QPD) are bounded well by $\bar{\mathcal{A}}^+(s)$ and $\bar{\mathcal{A}}(x)$ below and above (Theorem 7.8), and under stricter conditions, the predicted active set $\bar{\mathcal{A}}(x)$ is equivalent to $\mathcal{A}(x^*)$ (Theorem 7.9) and the predicted strongly active set $\bar{\mathcal{A}}^+(s)$ equivalent to $\mathcal{A}^+(s^*)$ (Theorem 7.10).

**Theorem 7.8.** *Let $C > 0$ and fix the vector of perturbations $\lambda$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{C}{8\max(\tau_p, \tau_d)}\right), \tag{7.23}$$

*where $\tau_p$ and $\tau_d$ are problem-dependent constants in (7.19). Let $(x, y, s) \in \mathcal{N}(\gamma, \lambda)$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \min\left(1, \left(\frac{C}{2\max(\tau_p, \tau_d)C_4}\right)^2\right), \tag{7.24}$$

*where $\mathcal{N}(\gamma, \lambda)$ is defined in (7.18), $\mu_\lambda$ in (5.4) and $C_4 > 0$, defined in (7.20), is a problem-dependent constant. Then there exists a solution $(x^*, y^*, s^*)$ of (QPD) such that*

$$\bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*) \subseteq \bar{\mathcal{A}}(x), \tag{7.25}$$

*where $\bar{\mathcal{A}}^+(s)$ and $\bar{\mathcal{A}}(x)$ are defined in (5.12), $\mathcal{A}^+(s^*)$ and $\mathcal{A}(x^*)$ in (2.28).*

**Proof.** We mimic the proof of Theorem 5.6. From the complementary condition in (2.27), it is straightforward to derive $\mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*)$. From $\|\lambda\| < 1$, $\mu_\lambda < 1$ and (7.19), we have $\|x - x^*\| \leq \tau_p C_4\sqrt{\mu_\lambda} + 4\tau_p\|\lambda\|$. This, (7.23), and (7.24) give us that when $i \in \mathcal{A}(x^*)$, $x_i^* = 0$ and $x_i \leq \tau_p C_4\sqrt{\mu_\lambda} + 4\tau_p\|\lambda\| < C$. Thus $\mathcal{A}(x^*) \subseteq \bar{\mathcal{A}}(x)$. Similarly, if $i \notin \mathcal{A}(x^*)$, we have $s_i^* = 0$ and then $s_i \leq \tau_d C_4\sqrt{\mu_\lambda} + 4\tau_d\|\lambda\| < C$, which implies $\bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}(x^*)$. □

**Theorem 7.9.** *Let*

$$\psi_p = \inf_{x^* \in QΩ^P} \min_{i \in \Theta(x^*)} x_i^*, \tag{7.26}$$

*where $QΩ^P$ is the solution set of the primal problem in (QPD), and $\Theta(s^*)$ is defined*

*in (2.28). Assume $\psi_p > 0$. Fix $C$ and $\lambda$ such that*

$$C = \frac{\psi_p}{2} \quad \text{and} \quad 0 < \|\lambda\| < \min\left(1, \frac{\psi_p}{16\max(\tau_p, \tau_d)}\right). \tag{7.27}$$

*Let $(x, y, s) \in \mathcal{N}(\gamma, \lambda)$ with $\mu_\lambda$ sufficiently small, namely*

$$\mu_\lambda < \min\left(1, \left(\frac{\psi_p}{4\max(\tau_p, \tau_d)C_4}\right)^2\right), \tag{7.28}$$

*where $\tau_p$ and $\tau_d$ are problem-dependent constants in (7.19), $\mathcal{N}(\gamma, \lambda)$ is defined in (7.18), $\mu_\lambda$ in (5.4) and $C_4$ in (7.20). Then there exists an optimal solution $(x^*, y^*, s^*)$ of (QPD), such that*

$$\bar{\mathcal{A}}(x) = \mathcal{A}(x^*),$$

*where $\bar{\mathcal{A}}(x)$ is defined in (5.12) and $\mathcal{A}(x^*)$ in (2.28).*

**Proof.** Setting $C = \frac{\psi_p}{2}$ in Theorem 7.8, we have (7.25). It remains to prove $\bar{\mathcal{A}}(x) \subseteq \mathcal{A}(x^*)$. If $i \notin \mathcal{A}(x^*)$, $i \in \Theta(x^*)$ and we have $x_i^* > 0$. Then from (7.26), (7.27) and (7.28), $x_i \geq x_i^* - \tau_p C_4\sqrt{\mu_\lambda} - 4\tau_p\|\lambda\| > \psi_p - \frac{\psi_p}{2} = C$, namely $i \notin \bar{\mathcal{A}}(x)$. Thus $\bar{\mathcal{A}}(x) \subseteq \mathcal{A}(x^*)$. □

**Theorem 7.10.** *Let*

$$\psi_d = \inf_{(y^*, s^*) \in Q\Omega^D} \min_{i \in \mathcal{A}^+(s^*)} s_i^*, \tag{7.29}$$

*where $Q\Omega^D$ is the solution set of the primal problem in (QPD), and $\mathcal{A}^+(s^*)$ is defined in (2.28). Assume $\psi_d > 0$. Fix $C$ and $\lambda$ such that*

$$C = \frac{\psi_d}{2} \quad \text{and} \quad 0 < \|\lambda\| < \min\left(1, \frac{\psi_d}{16\max(\tau_p, \tau_d)}\right). \tag{7.30}$$

*Let $(x, y, s) \in \mathcal{N}(\gamma, \lambda)$ with $\mu_\lambda$ sufficiently small, namely*

$$\mu_\lambda < \min\left(1, \left(\frac{\psi_d}{4\max(\tau_p, \tau_d)C_4}\right)^2\right), \tag{7.31}$$

*where $\tau_p$ and $\tau_d$ are problem-dependent constants in (7.19), $\mathcal{N}(\gamma, \lambda)$ is defined in (7.18), $\mu_\lambda$ in (5.4) and $C_4$ in (7.20). Then there exists an optimal solution $(x^*, y^*, s^*)$ of (QPD), such that*

$$\bar{\mathcal{A}}^+(s) = \mathcal{A}^+(s^*)$$

*where $\bar{\mathcal{A}}^+(s)$ is defined in (5.12) and $\mathcal{A}^+(s^*)$ in (2.28).*

**Proof.** Setting $C = \frac{\psi_d}{2}$ in Theorem 7.8, we have (7.25). It remains to prove that $\mathcal{A}^+(s^*) \subseteq \bar{\mathcal{A}}^+(s)$. If $i \in \mathcal{A}^+(s^*)$, we have $s_i^* > 0$. Then from (7.29), (7.30) and (7.31), $s_i \geq s_i^* - \tau_d C_4 \sqrt{\mu} - 4\tau_d \|\lambda\| > \psi_d - \frac{\psi_d}{2} = C$, namely $i \in \bar{\mathcal{A}}^+(s)$. Thus $\mathcal{A}^+(s^*) \subseteq \bar{\mathcal{A}}^+(s)$. □

**Remarks on Theorems 7.8–7.10.**

- The results for LP (Theorems 5.6 – 5.8) only require the primal-dual pair $(x, y, s)$ to be in the strictly feasible set of the perturbed problem, but we need to restrict $(x, y, s)$ to the symmetric neighbourhood defined in (7.18) for the QP case. This is a more restrictive condition but essential to the proof of Lemma 7.7. The presence of $\sqrt{\mu_\lambda}$ in (7.19) leads to a squared term in the thresholds (7.24), (7.28) and (7.31) for $\mu_\lambda$, which implies that, comparing with the results for LP, we may need to decrease $\mu_\lambda$ further before we can predict the optimal active set of a QP problem.
- Theorems 7.8 shows that the predicted strongly active set is included in the active set and the active set is a subset of the predicted active set. The intersection of these two predictions can serve as an approximation of the optimal active set, which is what we do in the implementation. Theorems 7.9 and 7.10 show that under certain conditions on the perturbations and duality gap, we could predict exactly the optimal active and strongly active sets at some optimal solution $(x^*, y^*, s^*)$ of (QPD). Similarly to the LP case, the same quantities $\psi_p$ and $\psi_d$ are present in the theorems. When (QPD) has a unique primal (dual) solution, $\psi_p > 0$ ($\psi_d > 0$). But $\psi_p$ and $\psi_d$ are only theoretical constants and our implementation does not depend on their values.

### 7.3.2 Predicting the original optimal tripartition

Let

$$\bar{\Theta}(x) = \{i \in \{1, \ldots, n\} \,|\, x_i \geq C\} \quad \text{and} \quad \bar{\mathcal{T}}(x, s) = \{1, \ldots, n\} \setminus (\bar{\mathcal{A}}^+(s) \cup \bar{\Theta}(x)), \tag{7.32}$$

where $C$ is some constant threshold and $\bar{\mathcal{A}}^+(s)$ is defined in (5.12). We consider $\big(\bar{\mathcal{A}}^+(s), \bar{\Theta}(x), \bar{\mathcal{T}}(x, s)\big)$ as the prediction of the optimal tripartition of (QPD) at the primal-dual pair $(x, y, s)$. Note that $\big(\bar{\mathcal{A}}^+(s), \bar{\Theta}(x), \bar{\mathcal{T}}(x, s)\big)$ may not be a tripartition for an arbitrary point as the complementary condition (7.2d) may not be satisfied and thus $\bar{\mathcal{A}}^+(s) \cap \bar{\Theta}(x)$ could be nonempty. The following two theorems, Theorems 7.11 and 7.12, show that, under certain conditions on $\mu_\lambda$ and $\lambda$, we are able to predict part or the whole of the tripartition.

**Theorem 7.11.** *Let $C > 0$ and fix the perturbation $\lambda$ such that $\|\lambda\|$ satisfies (7.23). Let $(x, y, s) \in \mathcal{N}(\gamma, \lambda)$ with $\mu_\lambda$ sufficiently small, namely, $\mu_\lambda$ satisfies (7.24). Then there exists an optimal solution $(x^*, y^*, s^*)$ of (QPD) such that*

$$\bar{\Theta}(x) \subseteq \Theta(x^*), \quad \bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}^+(s^*), \quad \text{and} \quad \mathcal{T}(x^*, s^*) \subseteq \bar{\mathcal{T}}(x, s), \tag{7.33}$$

*where $\Theta(x^*)$ and $\mathcal{A}^+(s^*)$ are defined in (2.28), $\mathcal{T}(x^*, s^*)$ in (2.30), $\bar{\Theta}(x)$ and $\bar{\mathcal{T}}(x, s)$ in (7.32), and $\bar{\mathcal{A}}^+(s)$ in (5.12).*

**Proof.** Theorem 7.8 shows that $\bar{\mathcal{A}}^+(s) \subseteq \mathcal{A}^+(s^*)$. From (7.25), we have $\mathcal{A}(x^*) \subseteq \bar{\mathcal{A}}(x)$. This, $\bar{\Theta}(x) = \{1, \ldots, n\} \setminus \bar{\mathcal{A}}(x)$, and $\Theta(x^*) = \{1, \ldots, n\} \setminus \mathcal{A}(x^*)$, give us that $\bar{\Theta}(x) \subseteq \Theta(x^*)$. $\mathcal{T}(x^*, s^*) \subseteq \bar{\mathcal{T}}(x, s)$ follows directly from (2.30) and (7.32). □

**Theorem 7.12.** *Let*

$$\psi = \min\left( \inf_{x^* \in Q\Omega^P} \min_{i \in \Theta(x^*)} x_i^*, \inf_{(y^*, s^*) \in Q\Omega^D} \min_{i \in \mathcal{A}^+(s^*)} s_i^* \right), \tag{7.34}$$

*where $Q\Omega^P$ is the solution set of the primal problem in (QPD), $Q\Omega^D$ is the solution set of the dual problem and $\Theta(s^*)$ and $\mathcal{A}^+(s^*)$ are defined in (2.28). Assume $\psi > 0$. Fix $C$ and $\lambda$ such that*

$$C = \frac{\psi}{2} \quad \text{and} \quad 0 < \|\lambda\| < \min\left(1, \frac{\psi}{16 \max(\tau_p, \tau_d)}\right). \tag{7.35}$$

*Let $(x, y, s) \in \mathcal{N}(\gamma, \lambda)$ with $\mu_\lambda$ sufficiently small, namely*

$$0 < \mu_\lambda < \min\left(1, \left(\frac{\psi}{4 \max(\tau_p, \tau_d) C_4}\right)^2\right), \tag{7.36}$$

*where $\tau_p$ and $\tau_d$ are problem-dependent constants in (7.19), $\mathcal{N}(\gamma, \lambda)$ is defined in (7.18), $\mu_\lambda$ in (5.4) and $C_4$ in (7.20). Then there exists an optimal solution $(x^*, y^*, s^*)$ of (QPD), such that*

$$\bar{\mathcal{A}}^+(s) = \mathcal{A}^+(s^*), \quad \bar{\Theta}(x) = \Theta(x^*) \quad \text{and} \quad \bar{\mathcal{T}}(x, s) = \mathcal{T}(x^*, s^*),$$

*where $\mathcal{T}(x^*, s^*)$ is defined in (2.30), $\bar{\mathcal{A}}^+(s)$ in (5.12), and $\bar{\Theta}(x)$ and $\bar{\mathcal{T}}(x, s)$ defined in (7.32).*

**Proof.** Setting $C = \frac{\psi}{2}$ in Theorem 7.11, we have (7.33). It remains to prove that $\Theta(x^*) \subseteq \bar{\Theta}(x)$ and $\mathcal{A}^+(s^*) \subseteq \bar{\mathcal{A}}^+(s)$. From (7.34), (7.35) and (7.36), if $i \in \Theta(x^*)$,

we have $x_i^* > 0$ and then $x_i \geq x_i^* - \tau_p C_4 \sqrt{\mu_\lambda} - 4\tau_p \|\lambda\| > \psi - \frac{\psi}{2} = C$, namely $i \in \bar{\Theta}(x)$. Thus $\Theta(x^*) \subseteq \bar{\Theta}(x)$. Similarly, we can also have $\mathcal{A}^+(s^*) \subseteq \bar{\mathcal{A}}^+(s)$. Therefore $\bar{\mathcal{T}}(x,s) = \mathcal{T}(x^*, s^*)$. □

### 7.3.3 Using the identification function as threshold

In Section 5.5, we have derived theoretical results of active-set prediction using the identification function for LP. In this section, we present similar results for QP. Given a strictly feasible point $(x, y, s) \in \mathcal{QF}_\lambda^0$ where $\mathcal{QF}_\lambda^0$ is defined in (7.1), we also denote $\rho(x, y, s)$ as the identification function for (QPD), namely

$$\rho(x, y, s) = \sqrt{r(x, s) + w(x, s)}, \tag{7.37}$$

where $r(x, s)$ and $w(x, s)$ are defined in (7.17). We employ the same notations as for LP for the predicted active sets, namely we denote $\hat{\mathcal{A}}(x, y, s)$ as the predicted primal active set and $\hat{\mathcal{A}}^+(x, y, s)$ the primal strongly active (dual inactive) set, where $\hat{\mathcal{A}}(x, y, s)$ is defined in (3.3) and $\hat{\mathcal{A}}^+(x, y, s)$ in (3.4).

**Lemma 7.13.** *Let $(x, y, s) \in \mathcal{QF}_\lambda^0$, where $\mathcal{QF}_\lambda^0$ is defined in (7.1) for some $\lambda \geq 0$. Then*

$$\rho^2(x, y, s) \leq C_4 \sqrt{\mu_\lambda} \max(\sqrt{\mu_\lambda}, 1) + 4\|\lambda\| \max(\|\lambda\|, 1), \tag{7.38}$$

*where $C_4$ is a problem-dependent constant defined in (7.20).*

**Proof.** The bound (7.38) follows from (7.21) and (7.22). □

Similar to Theorems 5.21 – 5.23, we derive the prediction results for QP, employing the identification function as the threshold. We only give a sketch of the proofs as they are similar to the proofs of Theorems 5.21 – 5.23.

**Theorem 7.14.** *Fix the vector of perturbations $\lambda$ such that*

$$0 < \|\lambda\| < \min\left(1, \frac{1}{8\max(\tau_p^2, \tau_d^2)}\right), \tag{7.39}$$

*where $\tau_p$ and $\tau_d$ are problem-dependent constants in (7.19). Let $(x, y, s) \in \mathcal{N}(\gamma, \lambda)$ with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \min\left(1, \left(\frac{1}{2C_4 \max(\tau_p^2, \tau_d^2)}\right)^2\right), \tag{7.40}$$

*where $\mathcal{N}(\gamma, \lambda)$ is defined in (7.18), $\mu_\lambda$ in (5.4) and $C_4 > 0$, defined in (7.20), is a problem-dependent constant. Then there exists a solution $(x^*, y^*, s^*)$ of (QPD)*

*such that*

$$\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*) \subseteq \mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x,y,s),$$

*where* $\hat{\mathcal{A}}(x,y,s)$ *is defined in* (3.3), $\hat{\mathcal{A}}^+(x,y,s)$ *in* (3.4), *and* $\mathcal{A}^+(s^*)$ *and* $\mathcal{A}(x^*)$ *in* (2.28).

**Proof.** From (7.38), (7.39) and (7.40), we have $\rho(x,y,s) < \frac{1}{\max(\tau_p,\tau_d)}$. This, (7.37) and (7.19) give us that, when $i \in \mathcal{A}(x^*)$, $x_i \leq \tau_p \rho^2(x,y,s) < \rho(x,y,s)$, namely $\mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x,y,s)$. Similarly we can have $\hat{\mathcal{A}}^+(x,y,s) \subseteq \mathcal{A}^+(s^*)$. □

**Theorem 7.15.** *Assume* $\psi_p > 0$, *where* $\psi_p$ *is defined in* (7.26). *Fix* $\lambda$ *such that*

$$0 < \|\lambda\| < \min\left(1, \frac{1}{8\max(\tau_p^2, \tau_d^2)}, \frac{\psi_p^2}{32}\right) \tag{7.41}$$

*where* $\tau_p$ *and* $\tau_d$ *are problem-dependent constants in* (7.19). *Let* $(x,y,s) \in \mathcal{N}(\gamma,\lambda)$ *with* $\mu_\lambda$ *sufficiently small, namely,*

$$\mu_\lambda < \min\left(1, \left(\frac{1}{2C_4\max(\tau_p^2,\tau_d^2)}\right)^2, \left(\frac{\psi_p^2}{8C_4}\right)^2\right), \tag{7.42}$$

*where* $\mathcal{N}(\gamma,\lambda)$ *is defined in* (7.18), $\mu_\lambda$ *in* (5.4) *and* $C_4 > 0$, *defined in* (7.20), *is a problem-dependent constant. Then there exists a solution* $(x^*,y^*,s^*)$ *of* (QPD) *such that*

$$\hat{\mathcal{A}}(x,y,s) = \mathcal{A}(x^*),$$

*where* $\hat{\mathcal{A}}(x,y,s)$ *is defined in* (3.3), *and* $\mathcal{A}(x^*)$ *in* (2.28).

**Proof.** From Theorem 7.14, we have $\mathcal{A}(x^*) \subseteq \hat{\mathcal{A}}(x,y,s)$ and so it remains to prove $\hat{\mathcal{A}}(x,y,s) \subseteq \mathcal{A}(x^*)$. (7.38), (7.41) and (7.42) give us that $\rho(x,y,s) < \frac{\psi_p}{2}$. From this, (7.37) and (7.19), we can derive, when $i \notin \mathcal{A}(x^*)$, $x_i^* \geq \psi_p - \tau_p\rho^2(x,y,s) > \rho(x,y,s)$. □

**Theorem 7.16.** *Assume* $\psi_d > 0$, *where* $\psi_d$ *is defined in* (7.29). *Fix* $\lambda$ *such that*

$$0 < \|\lambda\| < \min\left(1, \frac{1}{8\max(\tau_p^2, \tau_d^2)}, \frac{\psi_d^2}{32}\right) \tag{7.43}$$

*where* $\tau_p$ *and* $\tau_d$ *are problem-dependent constants in* (7.19). *Let* $(x,y,s) \in \mathcal{N}(\gamma,\lambda)$

*with $\mu_\lambda$ sufficiently small, namely,*

$$\mu_\lambda < \min\left(1, \left(\frac{1}{2C_4 \max(\tau_p^2, \tau_d^2)}\right)^2, \left(\frac{\psi_d^2}{8C_4}\right)^2\right), \tag{7.44}$$

*where $\mathcal{N}(\gamma, \lambda)$ is defined in (7.18), $\mu_\lambda$ in (5.4) and $C_4 > 0$, defined in (7.20), is a problem-dependent constant. Then there exists a solution $(x^*, y^*, s^*)$ of (QPD) such that*

$$\hat{\mathcal{A}}^+(x, y, s) = \mathcal{A}^+(s^*),$$

*where $\hat{\mathcal{A}}^+(x, y, s)$ is defined in (3.4), and $\mathcal{A}^+(s^*)$ in (2.28).*

**Proof.** From Theorem 7.14, we have $\hat{\mathcal{A}}^+(x, y, s) \subseteq \mathcal{A}^+(s^*)$ and so it remains to prove $\mathcal{A}^+(s^*) \subseteq \hat{\mathcal{A}}^+(x, y, s)$. We first get $\rho(x, y, s) < \frac{\psi_d}{2}$ from (7.38), (7.43) and (7.44). From this, (7.37) and (7.19), we can then show for any $i \in \mathcal{A}^+(s^*)$, $s_i \geq \psi_d - \tau_d \rho^2(x, y, s) > \rho(x, y, s)$. □

## 7.4 Numerical experiments for quadratic programming using perturbations

### 7.4.1 The perturbed algorithm and its implementation

We have given that the perturbed algorithm framework for LP in Algorithm 6.1. As we have mentioned earlier in Section 2.3, the main difference between IPMs for LP and QP is the Newton system that we need to solve in order to get the search directions; the latter has an extra matrix $H$. For clarity, we state the algorithm for QP in Algorithm 7.1.

**Algorithm without perturbations for QP.** For comparison in the numerical tests, we refer to the algorithm with no perturbations (Algorithm 7.1 with $\lambda = \phi = 0$) as **Algorithm 7.2**. We also use the same notation $\mu^k$ defined in (2.13) for the duality gap for Algorithm 7.2.

Most of the implementation details follow similarly to the LP case unless specified. We **shrink perturbations** according to the value of the smallest elements of the current iterate, for instance, at iteration $k$, we choose a fixed fraction of $\lambda^k$ when $\min(x^k) > 0$, otherwise we find a point on the line segment connecting $\lambda^k$ and $-\min(x^k)e$; similarly for $\phi^k$. The **initial perturbations** are set to $\lambda^0 = \phi^0 = 10^{-3}e$ for all numerical tests. We utilise the same **active-set prediction** procedure proposed in Procedure 3.2, namely, we move the indices between the predicted active, predicted inactive, and undetermined sets, depending on whether the criteria $x_i^k < C$ and $s_i^k > C$ are satisfied. **Termination criteria** will be defined for each set of tests. Relative residual is also employed in the following tests to measure the distance from the iterates to

**Algorithm 7.1** The Perturbed Algorithm with Active-set Prediction for QP

**Step 0:** choose perturbations $(\lambda^0, \phi^0) > 0$ and calculate a Mehrotra starting point $(x^0, y^0, s^0)$;[20]

**for** $k = 0, 1, 2, \ldots$ **do**

**Step 1:** solve the perturbed Newton system (7.5) using the augmented system approach, namely

$$\begin{bmatrix} -H - D_\lambda^{-2} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \end{bmatrix} = - \begin{bmatrix} R_d^k - (X^k + \Lambda^k)^{-1} R_{\mu_\lambda}^k \\ R_p^k \end{bmatrix},$$
$$\Delta s^k = - \left(X^k + \Lambda^k\right)^{-1} \left(R_{\mu_\lambda}^k + \left(S^k + \Phi^k\right) \Delta x^k\right),$$

where $D_\lambda = \left(S^k + \Phi^k\right)^{-\frac{1}{2}} \left(X^k + \Lambda^k\right)^{\frac{1}{2}}$, $R_p^k = Ax^k - b$, $R_d^k = A^T y^k + s^k - Hx^k - c$, $R_{\mu_\lambda}^k = \left(X^k + \Lambda^k\right)\left(S^k + \Phi^k\right) e - \sigma^k \mu_\lambda^k e$, and where $\sigma^k \in [0, 1]$ is defined in (2.15) and $\mu_\lambda^k$ in (6.1);

**Step 2:** compute possibly distinct stepsizes $\alpha_p^k$ for the primal iterates $\Delta x^k$ and $\alpha_d^k$ for the dual ones $(\Delta y^k, \Delta s^k)$ following (6.2);

**Step 3:** update $x^{k+1} = x^k + \alpha_p^k \Delta x^k$ and $(y^{k+1}, s^{k+1}) = (y^k, s^k) + \alpha_d^k (\Delta y^k, \Delta s^k)$;

**Step 4:** predict the optimal active set of (QPD) and denote as $\mathcal{A}^k$;

**Step 5:** terminate if some termination criterion is satisfied;

**Step 6:** calculate $(\lambda^{k+1}, \phi^{k+1})$ possibly by shrinking $(\lambda^k, \phi^k)$ so that $(x^{k+1} + \lambda^{k+1}, s^{k+1} + \phi^{k+1}) > 0$;

**end for**

the optimal solution set of ($\text{QPD}_\lambda$), namely

$$\text{QrelRes}_\lambda^k = \frac{|| \left(Ax^k - b, A^T y^k + s^k - Hx^k - c, \left(X^k + \Lambda^k\right)\left(S^k + \Phi^k\right) e\right) ||_\infty}{1 + \max\left(||b||_\infty, ||c||_\infty\right)}. \tag{7.45}$$

### 7.4.2 Test problems

**Randomly generated problems (QTS1).** We first randomly generate the number of constraints $m \in (10, 200)$, the number of variables $n \in (20, 500)$ and the matrix $A$ following the same procedure described in Section 3.4.2 for generating random LP test problems (TS1). Then randomly generate a full rank square matrix $B \in \mathbb{R}^{n \times n}$ and set the quadratic term $H = B'B$. Next we generate a triple $(x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ with $(x, s) \geq 0$ and density about 0.5. Finally we obtain $b = Ax$ and $c = A^T y + s - Hx$. Thus $(x, y, s)$ is used as a feasible point for this problem. 50 problems are generated for this test set.

**Randomly generated degenerate problems (QTS2).** First generate $m$, $n$, $A$ and $H$ as for QTS1. Apart from generating a feasible point as we do for QTS1, we generate a primal-dual degenerate optimal solution here. Namely we generate a triple

[20] We use the Mehrotra's starting point (see Section 2.2.2 on page 22) for (QPD) as the starting point for both Algorithms 7.1 and 7.2.

$(x, y, s)$ with $(x, s) \geq 0$, $x_i s_i = 0$ for all $i \in \{1, \ldots, n\}$ and the number of positive components of $x$ strictly less than $m$ and that of $s$ strictly less than $n - m$. Then we get $b$ and $c$ as for QTS1. 50 problems are also generated for this test set.

**Convex QP test problems from Netlib [44] and Maros and Meszaros' test sets [87] (QTS3).** We choose 7 small problems from the Netlib LP test set and add the identity matrix as the quadratic term. We also choose 13 small problems from Maros and Meszaros' convex QP collection[21]. All test problems have been transformed to the form with only equality constraints and nonnegative bounds on $x$ by adding slack variables. The dimensions of the problems are small, namely $m < 200$ and $n < 250$ including slack variables. For the full list of the problems, see Table 7.1. Note that the problems whose names start with 'QP_' are obtained from NETLIB.

Table 7.1: Convex QP test problems from Netlib and Maros and Meszaros' test set

| Name | m | n | Name | m | n |
|---|---|---|---|---|---|
| QP_ADLITTLE | 55 | 137 | QP_AFIRO | 27 | 51 |
| QP_BLEND | 74 | 114 | QP_SC50A | 49 | 77 |
| QP_SC50B | 48 | 76 | QP_SCAGR7 | 129 | 185 |
| QP_SHARE2B | 96 | 162 | CVXQP1_S | 150 | 200 |
| CVXQP2_S | 125 | 200 | CVXQP3_S | 175 | 200 |
| DUAL1 | 86 | 170 | DUAL2 | 97 | 192 |
| DUAL3 | 112 | 222 | DUAL4 | 76 | 150 |
| HS118 | 44 | 59 | HS21 | 3 | 5 |
| HS51 | 3 | 10 | HS53 | 8 | 10 |
| HS76 | 3 | 7 | ZECEVIC2 | 4 | 6 |

### 7.4.3 On the accuracy of optimal active-set predictions

The main task for this test is to compare the three prediction ratios proposed in Section 3.4.3 for Algorithms 7.1 and 7.2. To measure and compare the accuracy of the predicted active sets, we terminate Algorithms 7.1 and 7.2 at the same iteration, and compare the predicted active sets with the original optimal active set at a solution obtained from the active-set method and that at a maximal complementary solution (the analytic center of the solution set) from an interior point method.[22] These two original optimal active sets can be different.[23] Through this test, we also try to answer which active sets (at a solution from the active-set solver or a maximal complementary solution) Algorithm 7.1 predicts. We test on two test cases, random problems (QTS1) and random degenerate problems (QTS2).

[21] www.doc.ic.ac.uk/~im/#DATA or download .mat format from my collection https://github.com/YimingYAN/QP-Test-Problems.

[22] We solve the problem using Matlab's QP solver *quadprog* with the 'Algorithm' option set to interior point or active set and consider all variables of the optimal solution $x^*$ less than $10^{-5}$ as active.

[23] The difference is about 5% on average for problems in QTS1 and 30% for problems in QTS2.

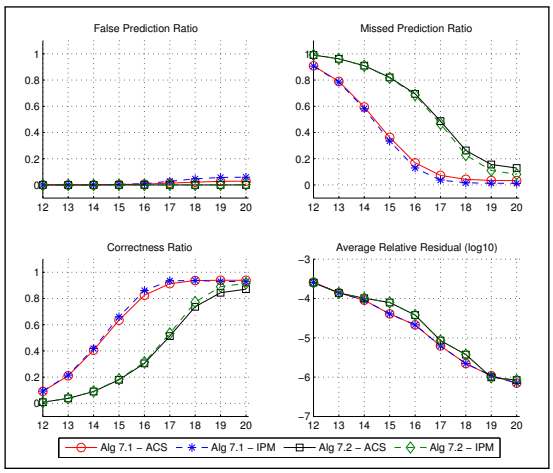



Figure 7.1: Prediction ratios for randomly generated QP problems

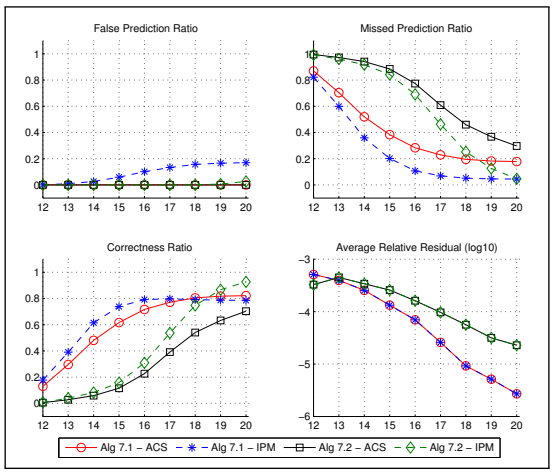



Figure 7.2: Prediction ratios for randomly generated primal-dual degenerate QP problems

In Figures 7.1 and 7.2, the x-axis gives the number of interior point iterations at which we terminate Algorithms 7.1 and 7.2 and the y-aixs shows the average value of corresponding measures. The first three plots (from top to bottom, left to right) present the corresponding prediction ratios. In each plot, we compare the predicted active set from Algorithm 7.1 with that from the active-set solver (the red solid line with circle), Algorithm 7.1 with the interior point solver (the blue dashed line with star sign), Algorithm 7.2 with the active-set solver (the black solid with square sign) and Algorithm 7.2 with the interior point solver (green dashed line with diamond sign). The last figure shows the log10-scaled average relative residuals (7.45) of Algorithms 7.1 or 7.2.

- Generally speaking, using perturbations yields earlier and better prediction of the original optimal active set for both test cases, in terms of the correctness ratios. Similar to the linear case, the correctness ratios from the perturbed algorithms are over two times higher than that from the unperturbed ones at some iterations, for test problems in both QTS1 and QTS2.

- It seems that the perturbed algorithm predicts the active set for an optimal solution of the original problem obtained from an active-set solver. Although it is not obvious for test problems in QTS1, the difference is much clearer for the degenerate case QTS2. In Figure 7.2, the false-prediction ratio for Algorithm 7.1 and the interior point solver is about 17% at the $20^{\text{th}}$ iteration but that for Algorithm 7.1 and the active-set solver stays close to 0.

- In Figure 7.2, we can also observe that after the $18^{\text{th}}$ iteration, the average correctness ratios comparing Algorithm 7.2 with the IPM solver are better than that comparing Algorithm 7.1 with active-set solver. This is because at the last few iterations the perturbations are not zero (on average about $\mathcal{O}(10^{-3})$) and can-

not shrink further; so the iterates of Algorithm 7.1 cannot keep moving closer to the original optimal solution, which prevents Algorithm 7.1 from improving the correctness ratios.

- Ultimately, the correctness ratio comparing Algorithm 7.2 with the interior point solver should go to 1 but then it would need to solve the problems to high accuracy ($10^{-8}$). As our implementation is for proof of concept, it can experience numerical issues when solving too far.

- Another interesting phenomenon is that the relative residual of Algorithm 7.1 seems to decrease faster than that of Algorithm 7.2. It suggests that using perturbations may help stabilise the Newton system and thus generate better search directions, especially for the degenerate problems in QTS2.

### 7.4.4 Solving the sub-problems

In this test, we first run Algorithm 7.1 and terminate it when $\mu_\lambda^k < 10^{-3}$, record the number of interior point iterations, remove zero variables and corresponding columns and/or rows of $H$, $A$ and $c$ from the original problem (QPD), and then solve the newly-formulated smaller-sized problem (sub-problem) using the active-set method. For comparison purposes we perform the same number of interior point iterations of Algorithm 7.2, predict the active set, formulate the sub-problem and solve it. We compare the number of active-set iterations used to solve the sub-problems from Algorithms 7.1 and 7.2.

It is also essential to make sure the sub-problems that we generate are equivalent to their original problems. Assume $\mathcal{A}^k$ is the predicted active set when terminating the interior point process at iteration $k$, $x^*_{\text{sub}}$ the optimal solution of the subproblems from the active-set solver and $x^*$ an optimal solution of the original problem. Let $\mathcal{A}_c^k = \{1, \ldots, n\} \setminus \mathcal{A}^k$ be the complement of $\mathcal{A}^k$. We consider the feasibility errors in the context of the original problem and the relative difference between the optimal objective values of the sub-problems and that of the original problems, namely,

- Feasibility error $= \dfrac{\|A_{\mathcal{A}_c^k} x^*_{\text{sub}} - b\|_\infty}{1 + \|b\|_\infty}$,
- Objective error $= \dfrac{|c^T_{\mathcal{A}_c^k} x^*_{\text{sub}} + \frac{1}{2}(x^*_{\text{sub}})^T H_{\mathcal{A}_c^k} x^*_{\text{sub}} - c^T x^* - \frac{1}{2}(x^*)^T H x^*|}{1 + |c^T x^* + \frac{1}{2}(x^*)^T H x^*|}$,
  where $H_{\mathcal{A}_c^k} = (H_{ij})_{i,j \in \mathcal{A}_c^k}$.

If the feasibility error is small, $\bar{x}^*$ with $\bar{x}^*_{\mathcal{A}^k} = 0$ and $\bar{x}^*_{\mathcal{A}_c^k} = x^*_{\text{sub}}$ is a feasible point for the original QP, and also optimal if the objective error is small as well.

**Randomly generated problems (QTS1 and QTS2).** Table 7.2 shows the average number of active-set iterations for the test problems in QTS1 and QTS2. It is clear that using perturbations saves a lot of active-set iterations, about 63% for problems

in QTS1 and 36% for QTS2. Though unfortunately degeneracy seems to disadvantage the improvement, it cannot cover the fact that using perturbations would enhance the capabilities of predicting a better active set of the original problem, in the context of primal-dual path-following interior point method structure, and potentially reduce the computational effort for solving a problem.

Table 7.2: Comparing the number of active-set iterations for Algorithms 7.1 and 7.2

| | Random problems | | Random degenerate problems | |
|---|---|---|---|---|
| | Algorithm 7.1 | Algorithm 7.2 | Algorithm 7.1 | Algorithm 7.2 |
| Avg # of active-set iters | 46 | 143 | 190 | 300 |
| Average $\mu_\lambda^k$ and $\mu^k$ when terminate IPM | $5.8 \times 10^{-04}$ | $8.0 \times 10^{-04}$ | $6.3 \times 10^{-04}$ | $7.8 \times 10^{-04}$ |

We check the objective and feasibility errors in Table 7.3. All optimal solutions of the sub-problems generated from Algorithms 7.1 and 7.2 are primal feasible for the original (QPD). For problems in QTS1, Algorithm 7.1 yields small average objective error, in the order of $10^{-7}$. For QTS2, the average error from Algorithm 7.2 is slightly higher, which is in the order of $10^{-6}$, but still acceptable, especially 90% of the test problems in QTS2 have small relative errors, in the order of $10^{-16}$ (can be considered as zero in MATLAB). This is, to some extend, even better than the result for the test case QTS1.

Table 7.3: Comparing the relative errors for Algorithms 7.1 and 7.2

| | Random problems | | Random degenerate problems | |
|---|---|---|---|---|
| | Algorithm 7.1 | Algorithm 7.2 | Algorithm 7.1 | Algorithm 7.2 |
| Average objective errors | $2.0 \times 10^{-07}$ | $9.2 \times 10^{-17}$ | $6.4 \times 10^{-06}$ | $8.9 \times 10^{-17}$ |
| $90^{th}$ percentile of relative errors | $4.9 \times 10^{-07}$ | $3.3 \times 10^{-16}$ | $6.2 \times 10^{-16}$ | $3.5 \times 10^{-16}$ |
| Average feasibility errors | $5.4 \times 10^{-14}$ | $5.9 \times 10^{-14}$ | $6.4 \times 10^{-14}$ | $8.2 \times 10^{-14}$ |

**QP problems from the Netlib and Maros and Meszaros' test sets (QTS3).** We also observe good numerical results for a small set of QP problems from Netlib and Maros and Meszaros' convex QP test set (QTS3). We summarise the results in Table 7.4. For these problems, we save almost 50% of active-set iterations and all optimal solutions of the sub-problems from Algorithm 7.1 are feasible and optimal for the original problems. For details, see Section A.2.

Table 7.4: Numerical results for solving sub-problems for test case QTS3

| | Algorithm 7.1 | Algorithm 7.2 |
|---|---|---|
| Avg # of active-set iters | 6 | 13 |
| Average $\mu_\lambda^k$ and $\mu^k$ when terminate IPM | $4.6 \times 10^{-04}$ | $6.4 \times 10^{-04}$ |
| Average relative errors | $1.1 \times 10^{-15}$ | $1.8 \times 10^{-15}$ |
| $90^{th}$ percentile of relative errors | $9.2 \times 10^{-16}$ | $9.9 \times 10^{-16}$ |
| Average feasibility errors | $9.6 \times 10^{-13}$ | $8.8 \times 10^{-13}$ |

### 7.4.5 Comparisons between cut-off and the identification function

In this section, we illustrate the numerical performance of the identification function for QP, by comparing with the cutoff.

Note that at the first few iterations, the value of the identification function is too large (larger than 1) and so the predictions at these iterations could be inaccurate. Thus if we start the active-set prediction procedure (Procedure 3.2) from the beginning of the IPM iterative process, we could encounter high false-prediction ratios. Therefore, we choose to start the prediction when the identification function is less than 0.1.[24]

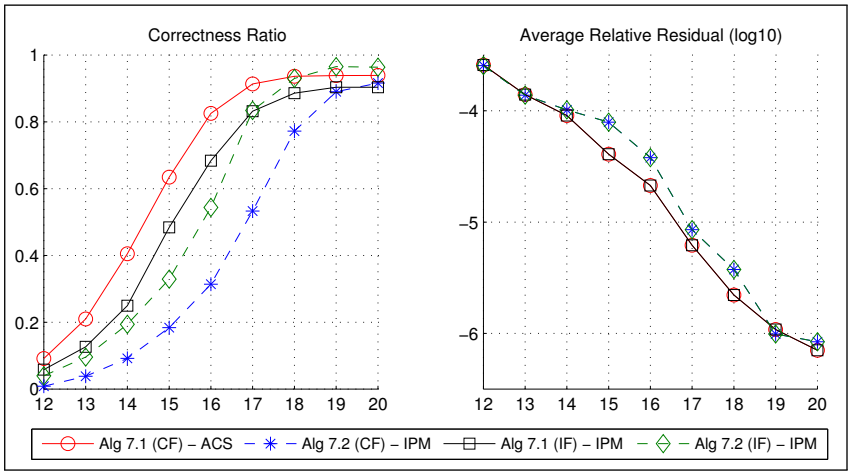



Figure 7.3: Comparing active-set prediction capabilities of cut-off and the identification function in the framework of Algorithm 7.1, for randomly generated problems in QTS1

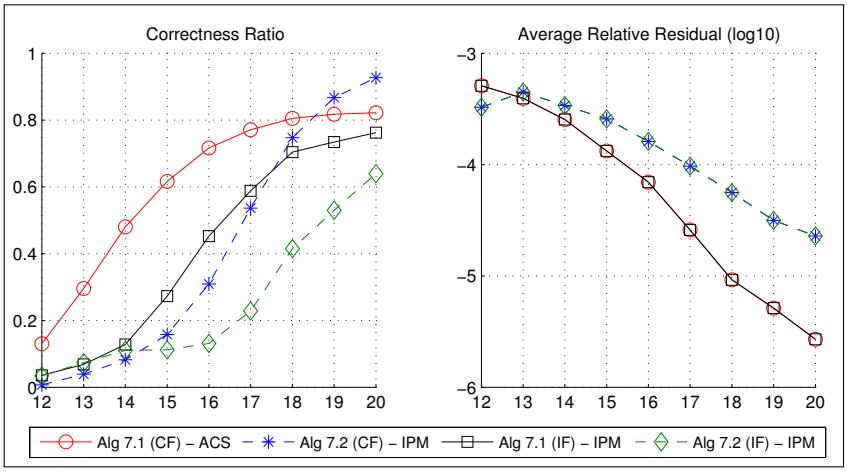



Figure 7.4: Comparing active-set prediction capabilities of cut-off and the identification function in the framework of Algorithm 7.1, for randomly generated primal-dual degenerate problems in QTS2

[24]This phenomenon is not observed for the LP test cases. And also when testing with cutoff, we start the prediction procedure from the beginning of the IPM iterative process, because we employ small value ($10^{-5}$) for cutoff.

Similarly to Algorithm 6.1 with the identification function, Algorithm 7.1 with the identification function predicts the optimal active set at an original maximal complementary solution obtained from the IPM solver, while Algorithm 7.1 with cut-off predicts the active set at an original optimal solution generated by the active-set solver. Thus in Figures 7.3 (for the random problems in QTS1) and 7.4 (for the random degenerate problems in QTS2), we present the correctness ratios by comparing the active set from Algorithm 7.1 using cut-off with that from MATLAB's active-set solver (solid red line with circle), Algorithm 7.2 using cut-off with MATLAB's IPM solver (dashed blue line with star), Algorithm 7.1 using the identification function with MATLAB's IPM solver (solid black line with square sign), and Algorithm 7.2 using the identification function with MATLAB's IPM solver (dashed green line with diamond sign).

- At early stages (average relative residuals less than $10^{-5}$), the average correctness ratios from the perturbed algorithm, Algorithm 7.1 with the identification function, is at least the same or slightly better than the unperturbed algorithm, Algorithm 7.2 with the identification function. However the gap is not so large as that for the LP case.

- Contrary to the LP case (Section 6.4.3), Algorithm 7.1 with cut-off performs slightly better than that with the identification function, for both test problems in QTS1 and QTS2.

- When using the unperturbed algorithm (Algorithm 7.2), it seems that employing the identification function can help improve the accuracy of active-set predictions for the random problems in QTS1, while this is not true for degenerate problems in QTS2.

## 7.5 Conclusions

Theoretically, we have extended the idea of active-set prediction using controlled perturbations from LP to QP. Numerically, we have obtained satisfactory preliminary results. Based on our observations, it seems that for the purpose of optimal active-set prediction for interior point methods for QP problems, the idea of using controlled perturbations is promising.

Note that our implementation of Algorithm 7.1 is preliminary. We have not employed techniques such as the predictor-corrector [93] or multiple centrality correctors [52]. Thus the algorithm may not be efficient enough and needs further refinement.

# 8 Conclusions and Future Directions

In this chapter, we summarise the main contributions of this thesis and present potential future directions of our research.

## 8.1 Summary and concluding remarks

In this thesis, we have studied existing techniques of optimal active-set predictions for interior point methods and proposed the idea of using controlled perturbations to enhance the capabilities of optimal active-set prediction for interior point methods for both LP and QP.

In Chapter 2, we have described the linear and quadratic programming problems, the structure of their solution set and the definition of the active set. We also presented the algorithm structure of an infeasible primal-dual path-following algorithm. Deriving the error bounds for linear and quadratic programming problems is the main contribution in this chapter. By formulating LP & QP as monotone LCP problems, we apply an existing LCP error bound. The error bounds derived in Section 2.4 can be used to bound the distance between an iterate and an optimal solution even if we do not know this solution, which serves as the foundation of our prediction results.

In Chapter 3, based on the error bound for LP obtained in Section 2.4, we have found an identification function for LP. We have also studied the best-known optimal active-set prediction techniques developed for IPMs, the indicators, and the simplest strategy, the cut-off. In the numerical tests, when we use the above mentioned three techniques to predict the optimal active set under a conservative procedure (Procedure 3.2), it seems that the identification function yields the best accuracy of prediction, followed by indicators and then cut-off. One common problem we have observed is that none of them seem able to predict a satisfyingly good proportion of the active constraints early enough in the IPM iterative process, which is an important challenge we would like to tackle in this thesis.

In Chapter 4, we have described the LP problems with controlled perturbations. We have also presented that under certain conditions, the optimal solution of the original problems lies on or in a neighbourhood of the central path of the perturbed problems.

Intuitively, when we solve the perturbed problems, we expect that the iterates may come close to the original optimal solution on the way to the perturbed optimal solution, which enables us to predict the original optimal active set before we solve the perturbed problems to optimality. Furthermore, we have discussed that under certain non-degeneracy assumptions, the perturbed problem has the same active set as the original one.

In Chapter 5, we have shown that our predicted active sets bound well the optimal active set of the original (unperturbed) LP. This result suggests a practical way to predict the optimal active set of the original problem, which is used in our implementations. Then we have also shown that our predicted set can exactly predict the original optimal active set under a certain nondegeneracy assumption. All these results make use of the error bound we derived for LP. At the end of this chapter, we have also presented conditions on problem conditioning that ensure our prediction of the optimal active set of the original LP can happen sooner than that of the optimal active set of the perturbed problems. This gives us hope that our approach may not need to solve the perturbed problems to high accuracy.

In Chapter 6, we have compared the accuracy of the optimal active-set predictions from the perturbed algorithm with that from the unperturbed one. We have observed that the perturbed algorithm with cut-off seems to predict the optimal active set at a vertex solution, while the perturbed algorithm with the identification function predicts the optimal active set at a strictly complementary solution. Nevertheless, the perturbed algorithm framework is at least as good and generally better, when compared with the unperturbed algorithm, in the context of optimal active-set prediction. We then conducted the crossover test, namely predict the optimal active set after some IPM iterations, generate an initial basis from that active set and run the simplex method from that initial basis. The perturbed algorithm also saves more simplex iterations. The numerical experiments have shown that our idea of using controlled perturbations to help predict the optimal active set is promising.

In Chapter 7, we have extended the idea from LP to convex QP, without assuming the existence of the strictly complementary solution. All major prediction results have been reproduced for QP, except the theorem that shows we could predict the original optimal active set sooner than we predict that of the perturbed problems. Additionally, we have proved that we can also predict the optimal tripartition of the original problems by solving the perturbed ones. Although our prototyped algorithm in MATLAB is not optimized or efficient enough and so we could not test large scale problems, the tests on random test problems and the small sized QP problems from NELTIB and Maros and Meszaros' convex QP test set did show some promising performance.

## 8.2 Future directions

There are several issues remaining for full validation of the proposed approach.

- **The choice of the initial perturbations**. We currently set the initial perturbations to a fixed small value that we then adjust, but they could be more suitably set to some problem-dependent value. In Theorem 5.7, the perturbations needed for an accurate prediction of the original optimal active set are bounded by some function of the problem conditioning. So we could attempt to relate the initial perturbations to the conditioning of the problem data $[A\ b]$.

- **Predictor-corrector and multiple centrality corrections**. Our current implementation simply follows the basic structure of path-following IPMs. Other variants of IPMs should be implemented and tested in the framework of controlled perturbations, with the aim of improving algorithm efficiency. For example Mehrotra's predictor-corrector method [93] can improve the numerical efficiency of IPMs often taking less iterations [83]; this variant of IPMs has been widely used in both academic and industrial softwares. Multiply centrality technique proposed by Gondzio [52] may also be a promising direction.

- **Other termination techniques.** Although our main aim here is to propose a method that can be used to predict the optimal active set as early as possible, it is equally important for us to make use of the predictions to terminate the algorithms. Currently we have tried crossover to the simplex method for LP and removing zero variables and corresponding columns/rows in problem data to reduce the problem dimension for QP. Other potential techniques are also worth being explored, for instance, the constraints reduction method [125], which uses the predicted active set to form a reduced version of normal equations so as to reduce the computational cost when calculating the Newton directions.

- **A large-scale implementation and testing** of the perturbed algorithm approach are needed to complete our numerical experiments for both LP and QP. Currently we prototyped our algorithms in MATLAB. A faster and more reliable version written in C++ is desirable. One of the core issues that affects the performance of the interior point method is the numeral linear algebra package that is used, especially in the factorisation routine. Fast and stable packages, such as MA57, may help improve the efficiency of the implementations. Eventually we expect to have an object-oriented package in C++ that handles large problems, for both LP and QP.

- **Extension to general nonlinear programming**. Interior point methods are also powerful tools for solving NLP problems and optimal active-set prediction techniques are equally, if not more, important to the algorithms for NLP. Thus it makes sense to explore how to use controlled perturbations to improve the active-set prediction capabilities for IPMs for NLP. An easier start may be to have a working code and run several tests on random or small NLP test problems, in order to understand the behaviours of the algorithms with perturbations, rather than to work directly on the theory.

- **Potential applications**. Active-set prediction techniques can be helpful in many different scenarios, such as solving mixed integer programming, finding active constraints in support vector machine, etc. All these may lead to potential applications of our perturbed approach. And it would be beneficial if we can demonstrate the use of our perturbed algorithms by solving some real world problems.

# Bibliography


[1] Clp – Coin-OR Linear Programming. http://projects.coin-or.org/Clp/.

[2] GLPK (GNU Linear Programming Kit). http://www.gnu.org/software/glpk, 2006.

[3] A. Altman. QHOPDM—a higher-order primal-dual method for large-scale convex quadratic programming. *European Journal of Operational Research*, 87(1):200–202, 1995.

[4] A. Altman and J. Gondzio. Regularized symmetric indefinite systems in interior point methods for linear and quadratic optimization. *Optimization Methods and Software*, 11(1-4):275–302, 1999.

[5] E. D. Andersen and Y. Ye. Combining interior-point and pivoting algorithms for linear programming. *Management Science*, 42(12):pp. 1719–1731, 1996.

[6] K. M. Anstreicher. Linear programming in $\mathcal{O}([n^3/ln\,n]L)$ operations. *SIAM Journal on Optimization*, 9(4):803–812, 1999.

[7] D. S. Atkinson and P. M. Vaidya. A cutting plane algorithm for convex programming that uses analytic centers. *Mathematical Programming*, 69(1-3):1–43, 1995.

[8] F. Babonneau, O. D. Merle, and J.-P. Vial. Solving large-scale linear multicommodity flow problems with an active set strategy and proximal-ACCPM. *Operations Research*, 54(1):184–197, 2006.

[9] F. Babonneau and J.-P. Vial. Accpm with a nonlinear constraint and an active set strategy to solve nonlinear multicommodity flow problems. *Mathematical programming*, 120(1):179–210, 2009.

[10] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali. Special simplex implementations and optimality conditions. In *Linear Programming and Network Flows*, pages 201–257. John Wiley & Sons, Inc., 2009.

[11] H. Y. Benson and D. F. Shanno. An exact primal—dual penalty method approach to warmstarting interior-point methods for linear programming. *Comput. Optim. Appl.*, 38(3):371–399, Dec. 2007.

[12] A. Berkelaar, B. Jansen, K. Roos, and T. Terlaky. Basis- and tripartition identification for quadratic programming and lineair complementary problem s; from an interior solution to an optimal basis and viceversa. Econometric Institute Report EI 9614-/A, Erasmus University Rotterdam, Econometric Institute, Jan. 1996.

[13] A. Berkelaar, K. Roos, and T. Terlaky. The optimal set and optimal partition approach to linear and quadratic programming. In T. Gal and H. Greenberg, editors, *Advances in Sensitivity Analysis and Parametic Programming*, volume 6 of *International Series in Operations Research & Management Science*, pages 159–202. Springer US, 1997.

[14] M. Berkelaar. lp solve: a mixed integer linear program solver, 1997.

[15] M. Berkelaar, K. Eikland, and P. Notebaert. *lpsolve : Open source (Mixed-Integer) Linear Programming system.*

[16] D. Bertsekas. Projected newton methods for optimization problems with simple constraints. In *Decision and Control including the Symposium on Adaptive Processes, 1981 20th IEEE Conference on*, volume 20, pages 762–767, Dec 1981.

[17] D. Bertsekas. *Nonlinear Programming.* Athena Scientific, 1999.

[18] R. E. Bixby, J. W. Gregory, I. J. Lustig, R. E. Marsten, and D. F. Shanno. Very large-scale linear programming: A case study in combining interior point and simplex methods. *Operations Research*, 40(5):885–897, 1992.

[19] R. E. Bixby and M. J. Saltzman. Recovering an optimal LP basis from an interior point solution. *Oper. Res. Lett.*, 15(4):169–178, May 1994.

[20] L. Blum, M. Shub, S. Smale, et al. On a theory of computation and complexity over the real numbers: *np*-completeness, recursive functions and universal machines. *Bulletin (New Series) of the American Mathematical Society*, 21(1):1–46, 1989.

[21] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, New York, NY, USA, 2004.

[22] J. Burke and J. Moré. On the identification of active constraints. *SIAM Journal on Numerical Analysis*, 25(5):1197–1211, 1988.

[23] S. L. Campbell and C. D. Meyer. *Generalized Inverses of Linear Transformations.* Society for Industrial and Applied Mathematics, 2009.

[24] C. Cartis and N. I. M. Gould. Finding a point in the relative interior of a polyhedron. Technical Report RAL 2006-016, Rutherford Appleton Laboratory, 2006.

[25] C. Cartis and Y. Yan. Active-set prediction for interior point methods using controlled perturbations. Technical Report ERGO-14-006, School of Mathematics, The University of Edinburgh, April 2014.

[26] R. W. Cottle, J.-S. Pang, and R. E. Stone. *The Linear Complementarity Problem.* SIAM, 2009.

[27] G. Dantzig. *Linear Programming and Extensions.* Princeton University Press, Aug. 1963.

[28] G. Dantzig and M. Thapa. *Linear Programming 2: Theory and Extensions*. Linear Programming 2: Theory and Extensions. U.S. Government Printing Office, 2003.

[29] R. De Leone and C. Lazzari. Error bounds for support vector machines with application to the identification of active constraints. *Optimization Methods Software*, 25(2):185–202, 2010.

[30] J. Dennis and R. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1996.

[31] I. I. Dikin. Iterative solution of problems of linear and quadratic programming. *Dokl. Akad. Nauk SSSR*, 174:747–748, 1967.

[32] A. S. El-Bakry, R. Tapia, and Y. Zhang. A study of indicators for identifying zero variables in interior point methods. *SIAM Review*, 36(1):45–72, 1994.

[33] A. Engau, M. F. Anjos, and A. Vannelli. A primal-dual slack approach to warmstarting interior-point methods for linear programming. In *Operations Research and Cyber-Infrastructure*, volume 47, pages 195–217. Springer US, 2009.

[34] A. Engau, M. F. Anjos, and A. Vannelli. On interior-point warmstarts for linear and combinatorial optimization. *SIAM Journal on Optimization*, 20(4):1828–1861, 2010.

[35] F. Facchinei, A. Fischer, and C. Kanzow. On the accurate identification of active constraints. *SIAM Journal on Optimization*, 9(2):14–32, 1998.

[36] M. C. Ferris, O. L. Mangasarian, and S. J. Wright. *Linear Programming with Matlab*. SIAM, 2007.

[37] A. V. Fiacco and G. P. McCormick. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1990.

[38] R. Fletcher. A general quadratic programming algorithm. *IMA Journal of Applied Mathematics*, 7(1):76–91, 1971.

[39] R. Fletcher. *Practical methods of optimization; (2nd ed.)*. Wiley-Interscience, New York, NY, USA, 1987.

[40] R. Freund. A potential-function reduction algorithm for solving a linear program directly from an infeasible "warm start". *Mathematical Programming*, 52(1-3):441–466, 1991.

[41] R. Freund. An infeasible-start algorithm for linear programming whose complexity depends on the distance from the starting point to the optimal solution. *Annals of Operations Research*, 62(1):29–57, 1996.

[42] R. M. Freund. Theoretical efficiency of a shifted-barrier-function algorithm for linear programming. *Linear Algebra and its Applications*, 152(0):19 – 41, 1991.

[43] K. R. Frisch. The logarithmic potential method of convex programming. Memorandum, University Institute of Economics, Oslo, May 1955.

[44] D. M. Gay. Electronic mail distribution of linear programming test problems. *Mathematical Programming Society COAL Newsletter*, 13:10–12, 1985.

[45] P. Gill, W. Murray, M. Saunders, J. Tomlin, and M. Wright. On projected newton barrier methods for linear programming and an equivalence to karmarkar's projective method. *Mathematical Programming*, 36(2):183–209, 1986.

[46] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for qpopt 1.0: a FORTRAN package for for quadratic programming. Technical Report SOL 95-4, Department of Operations Research, Stanford University, 1995.

[47] P. E. Gill and E. Wong. Methods for convex and general quadratic programming. *Numerical Analysis Report*, pages 10–1, 2013.

[48] J.-L. Goffin and J.-P. Vial. Cutting planes and column generation techniques with the projective algorithm. *Journal of Optimization Theory and Applications*, 65(3):409–429, 1990.

[49] D. Goldfarb. Extensions of newton's method and simplex methods for solving quadratic programs. *Numerical methods for non-linear optimization*, pages 239–254, 1972.

[50] A. J. Goldman and A. W. Tucker. Polyhedral convex cones. *Linear inequalities and related systems*, 38:19–40, 1956.

[51] J. Gondzio. Hopdm (version 2.12)—a fast lp solver based on a primal-dual interior point method. *European Journal of Operational Research*, 85(1):221–225, 1995.

[52] J. Gondzio. Multiple centrality corrections in a primal-dual method for linear programming. *Computational Optimization and Applications*, 6(2):137–156, 1996.

[53] J. Gondzio. Warm start of the primal-dual method applied in the cutting-plane scheme. *Math. Program.*, 83(1):125–143, Sept. 1998.

[54] J. Gondzio. Interior point methods 25 years later. *Eur. J. Oper. Res.*, 218(3):587–601, 2012.

[55] J. Gondzio. Matrix-free interior point method. *Computational Optimization and Applications*, 51(2):457–480, 2012.

[56] J. Gondzio, P. González-Brevis, and P. Munari. Large-scale optimization with the primal-dual column generation method. *ArXiv e-prints*, Sept. 2013.

[57] J. Gondzio and A. Grothey. Solving nonlinear portfolio optimization problems with the primal-dual interior point method. *European Journal of Operational Research*, 181(3):1019–1029, 2007.

[58] J. Gondzio and A. Grothey. A new unblocking technique to warmstart interior point methods based on sensitivity analysis. *SIAM Journal on Optimization*, 19(3):1184–1210, Nov. 2008.

[59] J. Gondzio and A. Grothey. Exploiting structure in parallel implementation of interior point methods for optimization. *Computational Management Science*, 6(2):135–160, 2009.

[60] J. Gondzio and J.-P. Vial. Warm start and $\epsilon$-subgradients in a cutting plane scheme for block-angular linear programs. *Computational Optimization and Applications*, 14(1):17–36, 1999.

[61] N. I. M. Gould, D. Orban, and P. L. Toint. Galahad, a library of thread-safe fortran 90 packages for large-scale nonlinear optimization. *ACM Trans. Math. Softw.*, 29(4):353–372, Dec. 2003.

[62] M. Grötschel and D. Mathematiker-Vereinigung. *Optimization Stories*. Dt. Mathematiker-Vereinigung, 2012.

[63] O. Güler, D. den Hertog, C. Roos, T. Terlaky, and T. Tsuchiya. Degeneracy in interior point methods for linear programming: a survey. *Annals of Operations Research*, 46-47:107–138, 1993. 10.1007/BF02096259.

[64] O. Güler and Y. Ye. Convergence behavior of interior-point algorithms. *Mathematical Programming*, 60(1-3):215–228, 1993.

[65] J. Hall. Towards a practical parallelisation of the simplex method. *Computational Management Science*, 7(2):139–170, 2010.

[66] D.-L. Han, J.-B. Jian, and J. Li. On the accurate identification of active set for constrained minimax problems. *Nonlinear Analysis: Theory, Methods & Applications*, 74(9):3022 – 3032, 2011.

[67] F. S. Hillier. *Introduction to operations research*. McGraw-Hill, eight edition, 2005.

[68] IBM. IBM ILOG CPLEX Optimizer. http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/, 2010.

[69] N. Karmarkar and K. Ramakrishnan. Computational results of an interior point algorithm for large scale linear programming. *Mathematical Programming*, 52(1-3):555–586, 1991.

[70] N. K. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[71] L. G. Khachiyan. Polynomial algorithms in linear programming. *Soviet Mathematics Doklady*, 20:191–194, 1979.

[72] V. Klee and G. J. Minty. How good is the simplex algorithm? In O. Shisha, editor, *Inequalities*, volume III, pages 159–175. Academic Press, New York, 1972.

[73] K. Koh, S. jean Kim, S. Boyd, and Y. Lin. An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine Learning Research*, 2007, 2007.

[74] M. Kojima, N. Megiddo, and S. Mizuno. A primal-dual infeasible-interior-point algorithm for linear programming. *Math. Program.*, 61(3):263–280, Sept. 1993.

[75] M. Kojima, S. Mizuno, and A. Yoshise. A polynomial-time algorithm for a class of linear complementarity problems. *Mathematical Programming*, 44:1–26, 1989. 10.1007/BF01587074.

[76] M. Kojima, S. Mizuno, and A. Yoshise. A primal-dual interior point algorithm for linear programming. *N. Megiddo, ed.,Progress in Mathematical Programming, Interior-Point and Related Methods (Springer, New York)*, pages 29–47, 1989.

[77] M. Lenard. A computational study of active set strategies in nonlinear programming with linear constraints. *Mathematical Programming*, 16(1):81–97, 1979.

[78] M. Lescrenier. Convergence of trust region algorithms for optimization with bounds when strict complementarity does not hold. *SIAM Journal on Numerical Analysis*, 28(2):476–495, 1991.

[79] A. Lewis and S. Wright. Identifying activity. *SIAM Journal on Optimization*, 21(2):597–614, 2011.

[80] R. Lewis and V. Torczon. Active set identification for linearly constrained minimization without explicit derivatives. *SIAM Journal on Optimization*, 20(3):1378–1405, 2010.

[81] LINDO Systems Inc. Lingo 14.0 - optimization modeling software for linear, nonlinear, and integer programming. http://www.lindo.com.

[82] I. Lustig, R. Marsten, and D. Shanno. Interior method vs. simplex method: Beyond netlib. *COAL Newsletter*, 19:41–44, 1991.

[83] I. Lustig, R. Marsten, and D. Shanno. On implementing mehrotra's predictor–corrector interior-point method for linear programming. *SIAM Journal on Optimization*, 2(3):435–449, 1992.

[84] I. J. Lustig, R. E. Marsten, and D. F. Shanno. Computational experience with a primal-dual interior point method for linear programming. *Linear Algebra and Its Applications*, 152:191–222, 1991.

[85] O. L. Mangasarian and J. Ren. New improved error bounds for the linear complementarity problem. *Mathematical Programming*, 66(2):241–255, 1994.

[86] H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, Mar. 1952.

[87] I. Maros and C. Mészáros. A repository of convex quadratic programming problems. *Optimization Methods and Software*, 11(1-4):671–681, 1999.

[88] B. A. McCarl, H. Moskowitz, and H. Furtan. Quadratic programming applications. *Omega*, 5(1):43–55, 1977.

[89] K. A. McShane, C. L. Monma, and D. Shanno. An implementation of a primal-dual interior point method for linear programming. *ORSA Journal on Computing*, 1(2):70–83, 1989.

[90] N. Megiddo. *Pathways to the optimal set in linear programming.* Springer, 1989.

[91] N. Megiddo. On finding primal- and dual-optimal bases. *ORSA Journal on Computing*, 3:63–65, 1991.

[92] S. Mehrotra. Finite termination and superlinear convergence in primal-dual methodsinite termination and superlinear convergence in primal-dual methods. Technical Report 91-13, Northwestern University, Evanston U.S.A., July 1991.

[93] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization*, 2(4):575–601, 1992.

[94] S. Mehrotra and Y. Ye. Finding an interior point in the optimal face of linear programs. *Mathematical Programming*, 62(1-3):497–515, 1993.

[95] J. E. Mitchell. *Karmarkar's algorithm and combinatorial optimization problems*. PhD thesis, Ithaca, NY, USA, 1988. AAI8900903.

[96] J. E. Mitchell. An interior point column generation method for linear programming using shifted barriers. *SIAM Journal on Optimization*, 4(2):423–440, 1994.

[97] J. E. Mitchell, K. Farwell, and D. Ramsden. Interior point methods for large-scale linear programming. In *Handbook of Optimization in Telecommunications*, pages 3–25. Springer, 2006.

[98] J. E. Mitchell and M. J. Todd. Solving combinatorial optimization problems using karmarkar's algorithm. *Mathematical Programming*, 56(1-3):245–284, 1992.

[99] S. Mizuno, M. J. Todd, and Y. Ye. On adaptive-step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research*, 18(4):pp. 964–981, 1993.

[100] R. D. Monteiro and I. Adler. Interior path following primal-dual algorithms. Part I: Linear programming. *Math. Program.*, 44(1):27–41, June 1989.

[101] R. D. Monteiro and I. Adler. Interior path following primal-dual algorithms. Part II: Convex quadratic programming. *Mathematical Programming*, 44(1-3):43–66, 1989.

[102] R. D. C. Monteiro and S. J. Wright. Local convergence of interior-point algorithms for degenerate monotone lcp. *Computational Optimization and Applications*, 3(2):131–155, 1994.

[103] J. Morales. A numerical study of limited memory BFGS methods. *Applied Mathematics Letters*, 15(4):481 – 487, 2002.

[104] J. Nanda, D. P. Kothari, and S. C. Srivastava. New optimal power-dispatch algorithm using Fletcher's quadratic programming method. *IEE Proceedings C (Generation, Transmission and Distribution)*, 136(3):153–161, 1989.

[105] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, 2006.

[106] C. Oberlin and S. Wright. Active set identification in nonlinear programming. *SIAM Journal on Optimization*, 17(2):577–605, 2006.

[107] K. A. Palaniswamy, J. K. Sharma, and K. B. Misra. Minimization of load curtailment in power system using quadratic programming. *Journal of the Institution of Engineers (India) Electrical Engineering Division*, 65:213–218, 1985.

[108] J.-S. Pang. Error bounds in mathematical programming. *Mathematical Programming*, 79(1):299–332, 1997.

[109] P. M. Pardalos and S. A. Vavasis. Quadratic programming with one negative eigenvalue is np-hard. *Journal of Global Optimization*, 1(1):15–22, 1991.

[110] R. Polyak. Modified barrier functions (theory and methods). *Mathematical Programming*, 54(1-3):177–222, 1992.

[111] F. A. Potra and S. J. Wright. Interior-point methods. *Journal of Computational and Applied Mathematics*, 124(1–2):281 – 302, 2000. Numerical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations.

[112] J. Renegar. A polynomial-time algorithm, based on newton's method, for linear programming. *Mathematical Programming*, 40(1-3):59–93, 1988.

[113] J. Renegar. *A mathematical view of interior-point methods in convex optimization*, volume 3. SIAM, 2001.

[114] C. Roos, T. Terlaky, and J. Vial. *Interior Point Methods for Linear Optimization*. Springer, 2006.

[115] SAS Institute Inc. SAS - Business Analytics and Business Intelligence Software. http://www.sas.com/.

[116] M. A. Saunders et al. Cholesky-based methods for sparse least squares: The benefits of regularization. *Linear and Nonlinear Conjugate Gradient-Related Methods*, pages 92–100, 1996.

[117] M. A. Saunders and J. A. Tomlin. *Solving regularized linear programs using barrier methods and KKT systems*, volume 10064. IBM Thomas J. Watson Research Center, 1996.

[118] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Chichester, 1986.

[119] G. Sierksma. *Linear and Integer Programming: Theory and Practice*. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, second edition, 2001.

[120] A. Skajaa, E. Andersen, and Y. Ye. Warmstarting the homogeneous and self-dual interior point method for linear and conic quadratic problems. *Mathematical Programming Computation*, 5:1–25, 2013.

[121] L. Sun, G. He, Y. Wang, and C. Zhou. An accurate active set newton algorithm for large scale bound constrained optimization. *Applications of Mathematics*, 56(3):297–314, 2011.

[122] R. Tapia and Y. Zhang. An optimal-basis identification technique for interior-point linear programming algorithms. *Linear Algebra and its Applications*, 152(0):343 – 363, 1991.

[123] R. Tapia, Y. Zhang, M. Saltzman, and A. Weiser. The mehrotra predictor-corrector interior-point method as a perturbed composite newton method. *SIAM Journal on Optimization*, 6(1):47–56, 1996.

[124] R. A. Tapia. Role of slack variables in quasi-newton methods for constrained optimization. Technical report, Rice Univ., Houston, TX (USA). Dept. of Mathematical Sciences, 1979.

[125] A. L. Tits, P.-A. Absil, and W. P. Woessner. Constraint reduction for linear programs with many inequality constraints. *SIAM J. Optim.*, 17(1):119–146, 2006.

[126] K. Tone. An active-set strategy in an interior point method for linear programming. *Mathematical Programming*, 59(1-3):345–360, 1993.

[127] R. J. Vanderbei. LOQO: an interior point code for quadratic programming. *Optimization Methods and Software*, 11(1-4):451–484, 1999.

[128] Z. Wang and Y.-x. Yuan. Componentwise error bounds for linear complementarity problems. *IMA Journal of Numerical Analysis*, 2009.

[129] P. J. Williams. *Effective finite termination procedures in interior-point methods for linear programming.* PhD thesis, Department of Computational and Applied Mathematics, Rice University, 1998.

[130] L. Winternitz, S. Nicholls, A. Tits, and D. O'Leary. A constraint-reduced variant of mehrotra's predictor-corrector algorithm. *Computational Optimization and Applications*, 51(3):1001–1036, 2012.

[131] M. Wright. Ill-conditioning and computational error in interior methods for nonlinear programming. *SIAM Journal on Optimization*, 9(1):84–111, 1998.

[132] S. Wright. Stability of augmented system factorizations in interior-point methods. *SIAM Journal on Matrix Analysis and Applications*, 18(1):191–222, 1997.

[133] S. Wright. Modified cholesky factorizations in interior-point algorithms for linear programming. *SIAM Journal on Optimization*, 9(4):1159–1191, 1999.

[134] S. J. Wright. Identifiable surfaces in constrained optimization. *SIAM J. Control Optim.*, 31(4):1063–1079, July 1993.

[135] S. J. Wright. *Primal-Dual Interior-Point Methods.* SIAM, 1997.

[136] N. Yamashita, H. Dan, and M. Fukushima. On the identification of degenerate indices in the nonlinear complementarity problem with the proximal point algorithm. *Mathematical Programming*, 99(2):377–397, 2004.

[137] Y. Ye. On the finite convergence of interior-point algorithms for linear programming. *Mathematical Programming*, 57(1):325–335, 1992.

[138] Y. Ye. *Interior point algorithms: theory and analysis*, volume 44. John Wiley & Sons, 2011.

[139] A. Yildirim and S. Wright. Warm-start strategies in interior-point methods for linear programming. *SIAM Journal on Optimization*, 12:782–810, 2000.

[140] Y. Zhang. Solving large-scale linear programs by interior-point methods under the matlab environment. *Optimization Methods and Software*, 10(1):1–31, 1998.

# A Appendix

## A.1 Results for crossover to simplex on selected Netlib problems

From the left to the right, we give the name of the test problems, number of equality constraints, number of variables, the value of duality gap $\mu_\lambda^K$ when we terminate the (perturbed) Algorithm 6.1, the value of duality gap $\mu^K$ when we terminate the (unperturbed) Algorithm 3.1, number of IPM iterations, the relative difference (see Footnote 16 on Page 84) between two bases generated from Algorithms 6.1 and 3.1, simplex iterations for Algorithm 6.1 and the simplex iterations for Algorithm 3.1. Since the algorithm without perturbations is terminated at the same IPM iteration as Algorithm 6.1, we show only the number of IPM iterations for the latter. Problems on which Algorithm 6.1 loses are marked in bold font. '—' means the simplex solver fails for a particular test problem.

Table A.1: Crossover to simplex test on a selection of Netlib problems.

| Probs | m | n | $\mu_\lambda^K$ | $\mu^K$ | IPM Itr | Basis Diff | splxItr Per | splxItr Unp |
|---|---|---|---|---|---|---|---|---|
| 25FV47 | 798 | 1854 | 9.38e-04 | 1.34e-03 | 35 | 0.15 | 4193 | 6951 |
| ADLITTLE | 55 | 137 | 3.79e-04 | 2.23e-04 | 16 | 0.45 | 18 | 119 |
| AFIRO | 27 | 51 | 3.68e-04 | 6.84e-06 | 11 | 0.07 | 9 | 9 |
| AGG3* | 516 | 758 | 9.05e-02 | 6.39e-02 | 25 | 0.07 | 112 | 123 |
| BLEND | 74 | 114 | 6.55e-04 | 7.21e-04 | 10 | 0.37 | 35 | 59 |
| BNL1 | 632 | 1576 | 5.41e-04 | 1.96e-02 | 28 | 0.31 | 1583 | 1632 |
| BRANDY | 149 | 259 | 4.83e-04 | 1.09e-03 | 18 | 0.38 | 76 | 278 |
| CZPROB | 737 | 3141 | 4.00e-04 | 1.67e-04 | 56 | 0.77 | 106 | 1822 |
| **E226** | 220 | 469 | 6.13e-04 | 6.98e-04 | 18 | 0.54 | 428 | 319 |
| FIT1D | 1050 | 2075 | 4.81e-04 | 2.00e-04 | 22 | 0.39 | 53 | 787 |
| FIT1P | 1026 | 2076 | 5.55e-04 | 4.04e-04 | 20 | 0.36 | 259 | 760 |
| FORPLAN* | 157 | 485 | 4.67e-03 | 1.33e-02 | 29 | 0.45 | 119 | 341 |
| **GROW7*** | 420 | 581 | 4.56e-02 | 5.56e-02 | 15 | 0.06 | 226 | 190 |
| **ISRAEL*** | 174 | 316 | 5.39e-02 | 1.87e-02 | 32 | 0.01 | 164 | 143 |
| **KB2** | 52 | 77 | 3.83e-04 | 1.15e-02 | 21 | 0.24 | 44 | 27 |
| SC50A | 49 | 77 | 1.64e-04 | 6.42e-05 | 10 | 0.12 | 22 | 27 |
| SC50B | 48 | 76 | 5.37e-04 | 1.59e-04 | 8 | 0.00 | 37 | 37 |
| SCAGR7 | 129 | 185 | 2.11e-04 | 2.80e-04 | 18 | 0.43 | 21 | 65 |
| SCFXM1 | 322 | 592 | 6.19e-04 | 4.35e-04 | 24 | 0.38 | 188 | 413 |
| **SCFXM2** | 644 | 1184 | 5.48e-04 | 1.04e-03 | 27 | 0.02 | 690 | 672 |
| SCFXM3 | 966 | 1776 | 8.73e-04 | 8.77e-04 | 28 | 0.01 | 1062 | 1074 |
| **SCRS8** | 485 | 1270 | 7.65e-04 | 1.42e-03 | 29 | 0.39 | 320 | 315 |
| SCSD1 | 77 | 760 | 5.54e-04 | 5.54e-04 | 7 | 0.95 | 125 | 214 |
| SCSD6 | 147 | 1350 | 5.86e-04 | 5.91e-04 | 8 | 0.94 | 346 | 411 |
| SCSD8 | 397 | 2750 | 4.88e-04 | 5.10e-04 | 11 | 0.90 | 366 | 965 |

Continued on next page

**Table A.1 – continued from previous page**

| Probs | m | n | $\mu_\lambda^K$ | $\mu^K$ | IPM Itr | Basis Diff | splxItr Per | splxItr Unp |
|---|---|---|---|---|---|---|---|---|
| SCTAP1 | 300 | 660 | 5.31e-04 | 3.64e-03 | 19 | 0.31 | 114 | 179 |
| SCTAP2 | 1090 | 2500 | 6.81e-04 | 2.30e-07 | 21 | 0.43 | 145 | 344 |
| SCTAP3 | 1480 | 3340 | 7.81e-04 | 1.11e-07 | 22 | 0.48 | 54 | 451 |
| SEBA | 1029 | 1550 | 5.92e-04 | 3.09e-04 | 23 | 0.04 | 43 | 70 |
| SHARE1B* | 112 | 248 | 2.87e-03 | 8.18e-02 | 27 | 0.24 | 176 | 204 |
| SHARE2B | 96 | 162 | 3.19e-04 | 3.90e-04 | 14 | 0.41 | 57 | 126 |
| SHIP04L | 356 | 2162 | 6.55e-04 | 2.92e-04 | 27 | 0.61 | 13 | 215 |
| SHIP08L | 688 | 4339 | 6.55e-04 | 5.41e-04 | 29 | 0.81 | 441 | 1056 |
| SHIP08S | 416 | 2171 | 6.34e-04 | 3.56e-04 | 26 | 0.76 | 70 | — |
| SHIP12S | 466 | 2293 | 2.12e-04 | 2.74e-05 | 33 | 0.71 | 18 | 541 |
| STAIR | 362 | 544 | 6.56e-04 | 1.11e-02 | 16 | 0.29 | 292 | 294 |
| **STOCFOR2** | 2157 | 3045 | 5.63e-04 | 4.74e-05 | 39 | 0.08 | 1213 | 796 |

## A.2 Numerical results for solving sub-problems

In Table A.2, from the left to the right, we present the name the problem, the number of equality constraints and variables, the value of duality gap when terminate Algorithms 7.1 and 7.2, the number of active-set iterations for solving the subproblems generated from Algorithms 7.1 and 7.2, the primal feasibility errors for the optimal solutions of the subproblems from Algorithms 7.1 and 7.2, and the objective errors between the subproblem and the original problem.

Table A.2: Solving sub-problem test on a selection of Netlib and Maros and Meszaros' convex QP problems.

| **Probs** | **m** | **n** | $\mu_\lambda^K$ | $\mu^K$ | **IPM Itr** | **actvItr Per** | **actvItr Unp** | **feaErr Per** | **feaErr Unp** | **relObjErr Per** | **relObjErr Unp** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| QP_ADLITTLE | 55 | 137 | 7.9e-04 | 9.6e-04 | 13 | 3 | 22 | 1.5e-12 | 1.0e-12 | 0.0e+00 | 1.6e-16 |
| QP_AFIRO | 27 | 51 | 1.9e-04 | 2.7e-04 | 13 | 1 | 5 | 2.9e-13 | 3.3e-13 | 7.2e-16 | 4.3e-16 |
| QP_BLEND | 74 | 114 | 2.9e-04 | 3.2e-04 | 14 | 7 | 38 | 5.4e-13 | 4.8e-13 | 4.8e-16 | 9.9e-15 |
| QP_SC50A | 49 | 77 | 9.2e-05 | 1.5e-04 | 10 | 1 | 1 | 2.6e-13 | 2.6e-13 | 7.5e-16 | 7.5e-16 |
| QP_SC50B | 48 | 76 | 5.3e-04 | 7.9e-04 | 8 | 2 | 3 | 3.2e-13 | 3.9e-13 | 1.2e-16 | 5.9e-16 |
| QP_SCAGR7 | 129 | 185 | 8.6e-04 | 1.3e-03 | 15 | 1 | 10 | 1.0e-11 | 9.5e-12 | 2.4e-16 | 2.4e-16 |
| QP_SHARE2B | 96 | 162 | 1.2e-04 | 1.4e-04 | 20 | 4 | 12 | 6.0e-12 | 4.7e-12 | 1.6e-14 | 2.3e-14 |
| CVXQP1_S | 150 | 200 | 4.5e-04 | 7.8e-04 | 8 | 1 | 16 | 6.0e-14 | 6.1e-14 | 1.6e-16 | 1.6e-16 |
| CVXQP2_S | 125 | 200 | 6.5e-04 | 1.1e-03 | 8 | 1 | 48 | 3.5e-14 | 4.0e-14 | 9.2e-16 | 4.6e-16 |
| CVXQP3_S | 175 | 200 | 5.4e-04 | 6.6e-04 | 9 | 2 | 4 | 6.3e-14 | 5.4e-14 | 4.7e-16 | 4.7e-16 |
| DUAL1 | 86 | 170 | 5.2e-04 | 6.1e-04 | 2 | 29 | 29 | 6.5e-15 | 6.5e-15 | 0.0e+00 | 0.0e+00 |
| DUAL2 | 97 | 192 | 5.1e-04 | 6.4e-04 | 2 | 5 | 5 | 6.4e-15 | 6.4e-15 | 0.0e+00 | 0.0e+00 |
| DUAL3 | 112 | 222 | 5.9e-04 | 6.1e-04 | 3 | 15 | 15 | 1.3e-14 | 1.3e-14 | 0.0e+00 | 0.0e+00 |
| DUAL4 | 76 | 150 | 3.0e-04 | 4.3e-04 | 4 | 14 | 14 | 1.3e-14 | 1.3e-14 | 0.0e+00 | 0.0e+00 |
| HS118 | 44 | 59 | 1.8e-04 | 2.8e-04 | 8 | 0 | 15 | 2.3e-14 | 1.5e-13 | 3.2e-16 | 0.0e+00 |
| HS21 | 3 | 5 | 3.4e-04 | 6.6e-04 | 10 | 2 | 2 | 5.2e-14 | 5.2e-14 | 0.0e+00 | 0.0e+00 |
| HS51 | 3 | 10 | 9.2e-04 | 7.3e-04 | 3 | 20 | 20 | 3.1e-15 | 3.1e-15 | 0.0e+00 | 0.0e+00 |
| HS53 | 8 | 10 | 9.9e-04 | 1.9e-03 | 6 | 1 | 1 | 2.2e-14 | 2.2e-14 | 0.0e+00 | 0.0e+00 |
| HS76 | 3 | 7 | 7.9e-05 | 1.5e-04 | 6 | 1 | 3 | 8.9e-16 | 1.9e-15 | 1.6e-16 | 0.0e+00 |
| ZECEVIC2 | 4 | 6 | 2.6e-04 | 4.0e-04 | 5 | 1 | 2 | 4.4e-15 | 3.9e-15 | 8.7e-16 | 0.0e+00 |
| Average: | | | 4.6e-04 | 6.4e-04 | 8 | 6 | 13 | 9.6e-13 | 8.6e-13 | 1.1e-15 | 1.8e-15 |
| 90th Pctl: | | | | | | | | 6.0e-12 | 4.7e-12 | 9.2e-16 | 9.9e-15 |


130