8-2016

Solution Techniques for Classes of Biobjective and Parametric Programs

Nathan Adelgren
Clemson University

Recommended Citation

Adelgren, Nathan, "Solution Techniques for Classes of Biobjective and Parametric Programs" (2016). *All Dissertations*. 1754.
https://tigerprints.clemson.edu/all_dissertations/1754

Solution Techniques for Classes of Biobjective and Parametric Programs

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Mathematical Sciences

by
Nathan Adelgren
August 2016

Accepted by:
Dr. Akshay Gupte, Committee Co-Chair
Dr. Margaret M. Wiecek, Committee Co-Chair
Dr. Pietro Belotti
Dr. Christopher Cox
Dr. Matthew Saltzman

Abstract

Mathematical optimization, or mathematical programming, has been studied for several decades. Researchers are constantly searching for optimization techniques which allow one to determine the ideal course of action in extremely complex situations. This line of scientific inquiry motivates the primary focus of this dissertation — nontraditional optimization problems having either multiple objective functions or parametric input. Utilizing multiple objective functions allows one to account for the fact that the decision process in many real-life problems in engineering, business, and management is often driven by several conflicting criteria such as cost, performance, reliability, safety, and productivity. Additionally, incorporating parametric input allows one to account for uncertainty in models' data, which can arise for a number of reasons, including a changing availability of resources, estimation or measurement errors, or implementation errors caused by storing data in a fixed precision format. However, when a decision problem has either parametric input or multiple objectives, one cannot hope to find a single, satisfactory solution. Thus, in this work we develop techniques which can be used to determine sets of desirable solutions. The two main problems we consider in this work are the biobjective mixed integer linear program (BOMILP) and the multiparametric linear complementarity problem (mpLCP).

BOMILPs are optimization problems in which two linear objectives are optimized over a polyhedron while restricting some of the decision variables to be integer. We present a new data structure in the form of a modified binary tree that can be used to store the solution set of BOMILP. Empirical evidence is provided showing that this structure is able to store these solution sets more efficiently than other data structures that are typically used for this purpose. We also develop a branch-and-bound (BB) procedure that can be used to compute the solution set of BOMILP. Computational experiments are conducted in order to compare the performance of the new BB procedure with current state-of-the-art methods for determining the solution set of BOMILP. The results provide strong evidence of the utility of the proposed BB method.

We also present new procedures for solving two variants of the mpLCP. Each of these procedures consists of two phases. In the first phase an initial feasible solution to mpLCP which satisfies certain criteria is determined. This contribution alone is significant because the question of how such an initial solution could be generated was previously unanswered. In the second phase the set of feasible parameters is partitioned into regions such that the solution of the mpLCP, as a function of the parameters, is invariant over each region. For the first variant of mpLCP, the worst-case complexity of the presented procedure matches that of current state-of-the-art methods for nondegenerate problems and is lower than that of current state-of-the-art methods for degenerate problems. Additionally, computational results show that the proposed procedure significantly outperforms current state-of-the-art methods in practice. The second variant of mpLCP we consider was previously unsolved. In order to develop a solution strategy, we first study the structure of the problem in detail. This study relies on the integration of several key concepts from algebraic geometry and topology into the field of operations research. Using these tools we build the theoretical foundation necessary to solve the mpLCP and propose a strategy for doing so. Experimental results indicate that the presented solution method also performs well in practice.

Dedication

To my wonderful wife Ashley. You are my motivation and inspiration.

Acknowledgments

First, I want to thank my Ph.D. advisors, Dr. Margaret M. Wiecek and Dr. Akshay Gupte, for their constant support, guidance, and encouragement. The lessons they have taught me extend well beyond academics; they have shaped the person I am.

I would also like to thank Dr. Pietro Belotti for his inspiration and support during my first year of academic research and for introducing me to the Linux operating system and quality programming; Dr. Matthew Saltzman for his advice and guidance on several occasions; Dr. Christopher Cox for introducing me to Clemson and for his encouragement throughout my studies; and Dr. Michael Burr for providing significant contributions to my research.

Additionally, I am extremely grateful to the following faculty members from institutions other than Clemson University who provided support, encouragement and inspiration during my early academic career: Dr. Justin Conroy, Dr. Erica Snow Simoson, and Dr. Lan Cheng of the State University of New York at Fredonia; and Dr. Fredrick Adkins and Dr. Yu-Ju Kuo of Indiana University of Pennsylvania.

I would also like to express my sincere gratitude to the Office of Naval Research for partially funding my research through grant number N00014-16-1-2725.

Finally, I want to thank my parents, Brian and Donna Adelgren, for believing in me and supporting me in this endeavor. They raised me to understand the importance of hard work and self discipline and always encouraged me to perform every task to the best of my ability. I would not be where I am today without their selfless generosity or the many sacrifices they made on my behalf.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Multiobjective programs (MOP) are optimization problems in which several objective functions are considered. Decision makers are constantly confronted with scenarios in which they are forced to consider multiple, conflicting objectives. When purchasing a vehicle, a person's choice is not often motivated by cost alone, but also by factors such as fuel economy, comfort, and safety. When building a stock portfolio, one is interested in achieving a high return without making many high risk investments. These are just a few examples of decision problems which can be formulated as multiobjective programs. Important applications of MOP arise in a variety of disciplines, including engineering, economics, logistics, and health care. When attempting to solve a MOP, conflict in the objectives typically prevents one from finding a single solution which simultaneously optimizes every objective. For this reason, it is generally accepted that to solve a MOP one must identify the set of so-called efficient solutions which offer the "best" compromises between the objectives. This is a significantly challenging problem, and there are many areas of multiobjective programming in which substantial improvements can still be made. Although many of the topics discussed in this dissertation offer contributions that extend well beyond the scope of MOP, the study of MOP has served as the primary motivation for the entirety of the work presented here. We now provide a brief overview of multiobjective optimization in which we introduce several of the key concepts that will be used throughout this work.

1.1 Overview of Multiobjective Optimization

In general, MOPs have the form

$$\begin{array}{ll} \min\limits_{x} & g(x) := [g_1(x), \ldots, g_\ell(x)] \\ \text{s.t.} & \qquad x \in \mathcal{X} \end{array} \tag{1.1}$$

where $g_1(\cdot), \ldots, g_\ell(\cdot)$ are functions of x, and $\mathcal{X} \subseteq \mathbb{R}^k$ is a *feasible set.* For each MOP, we also define

$$\Psi := \{\psi \in \mathbb{R}^\ell : \psi = g(x), x \in \mathcal{X}\}, \tag{1.2}$$

the set of all points in $\mathbb{R}^\ell$ which can be formed using the objective function values of feasible solutions to (1.1). We refer to the space $\mathbb{R}^\ell$ containing Ψ as the *objective space.*

Unlike single-objective programs, in general, a single solution to a multiobjective program which optimizes all objectives simultaneously does not exist. This is due to the fact that in most cases these objectives arise from conflicting criteria such as cost, performance, reliability, safety, and productivity. B a single optimal solution cannot be found, a set of efficient solutions which offer an acceptable compromise between the objectives is sought. In order to determine what types of solutions are "acceptable," we introduce several definitions and propositions.

Definition 1.1. Given distinct $\overline{x}, x' \in \mathcal{X}$, we say that $g(\overline{x})$ *dominates* $g(x')$ if $g(\overline{x}) \leq g(x')$. This dominance is *strong* if $g(\overline{x}) < g(x')$ and *weak* if $g(\overline{x}) = g(x')$.

Since it is generally unlikely that Ψ is known in its entirety, it is often of interest to consider particular subsets of Ψ.

Definition 1.2. Given $\Psi' \subseteq \Psi$ we say that $\psi' \in \Psi'$ is *nondominated* in Ψ' if there does not exist $\psi'' \in \Psi'$ such that ψ'' dominates ψ'.

Definition 1.3. A point $\overline{x} \in \mathcal{X}$ is *(weakly) efficient* if there does not exist $x' \in \mathcal{X}$ such that $g(x')$ (strongly) dominates $g(\overline{x})$.

The set of all efficient solutions to a given problem is denoted by attaching a subscript E to the symbol denoting the feasible set of the problem. Thus, the sets of efficient points for (1.1) is denoted $\mathcal{X}_E$.

Definition 1.4. A point $\overline{\psi} = g(\overline{x})$ is called *Pareto optimal* if $\overline{x} \in \mathcal{X}_E$.

Problem (1.1) is considered solved when $\mathcal{X}_E$ is found along with the corresponding set of Pareto optimal points

$$\Psi_P := \{\psi \in \mathbb{R}^\ell : \psi = g(x), x \in \mathcal{X}_E\}. \tag{1.3}$$

Definition 1.5. Given an instance of MOP, the *weighted sum problem* is the single objective program

$$\begin{array}{ll} \min\limits_{x} & h(x) := \sum\limits_{i=1}^{\ell} \lambda_i g_i(x) \\ \text{s.t.} & \quad x \in \mathcal{X} \end{array} \tag{1.4}$$

where $\lambda_i > 0$ for all $i \in \{1, \ldots, \ell\}$ and $\sum\limits_{i=1}^{\ell} \lambda_i = 1$.

For each $\lambda \in \mathbb{R}^\ell$ such that $\lambda_i > 0$ for all $i \in \{1, \ldots, \ell\}$ and $\sum\limits_{i=1}^{\ell} \lambda_i = 1$, we denote the optimal solution to (1.4) as x^*_λ.

Proposition 1.6. For each $\lambda \in \mathbb{R}^\ell$ such that $\lambda_i > 0$ for all $i \in \{1, \ldots, \ell\}$ and $\sum\limits_{i=1}^{\ell} \lambda_i = 1$, the solution x^*_λ is efficient for (1.1).

The claim of Proposition 1.6 is a well known result and its proof can be found, for example, in [27].

Definition 1.7. An efficient solution $\hat{x} \in \mathcal{X}_E$ for (1.1) is said to be *supported* if there exists $\lambda \in \mathbb{R}^\ell$ such that $\lambda_i > 0$ for all $i \in \{1, \ldots, \ell\}$, $\sum\limits_{i=1}^{\ell} \lambda_i = 1$ and $\hat{x} = x^*_\lambda$. Otherwise it is *unsupported*.

The definitions and propositions we have now introduced will serve as a sufficient foundation for the concepts we discuss throughout the rest of this work. We now move forward and consider specific subclasses of (1.1) in which integer decision variables are present.

1.2 Multiobjective Optimization Problems Involving Integer Decision Variables

It is often the case that in order to appropriately model a real-world scenario, a certain subset of the model's decision variables needs to be described using discrete quantities. This leads to two classes of problems which are quite worth studying and are the focus of the majority of this work: (i) multiobjective integer programs (MOIP), in which all decision variables take on discrete values, and (ii) multiobjective mixed-integer programs (MOMIP), in which a subset of the decision variables takes on discrete values. Most of the literature concerning these types of problems focuses on MOIP rather than MOMIP. For this reason there is still much research to be done in the area of MOMIP. Therefore, for the remainder of this work we consider two subclasses of MOMIP:

1. Multiobjective mixed-integer linear programs (MOMILP), i.e., MOMIPs of the form:

$$\begin{array}{ll} \min\limits_{x,y} & \left[f_1(x,y) = c_1^\top x + d_1^\top y, \ldots, f_\ell(x,y) = c_\ell^\top x + d_\ell^\top y\right] \\ \text{s.t.} & Ax + By \leq b \\ & x \in \mathbb{R}^m \\ & y \in \mathbb{Z}^n \end{array} \tag{1.5}$$

   where $A \in \mathbb{R}^{k\times m}$, $B \in \mathbb{R}^{k\times n}$, $b \in \mathbb{R}^k$ and $c_i \in \mathbb{R}^m$ and $d_i \in \mathbb{R}^n$ for each $i \in \{1, \ldots, \ell\}$.

2. Multiobjective mixed-integer quadratic programs (MOMIQP), i.e., MOMIPs of the form:

$$\begin{array}{ll} \min\limits_{x,y} & \left[f_1(x,y) = \frac{1}{2}\begin{bmatrix}x\\y\end{bmatrix}^\top Q_1 \begin{bmatrix}x\\y\end{bmatrix} + p_1^\top \begin{bmatrix}x\\y\end{bmatrix}, \ldots, f_\ell(x,y) = \frac{1}{2}\begin{bmatrix}x\\y\end{bmatrix}^\top Q_\ell \begin{bmatrix}x\\y\end{bmatrix} + p_\ell^\top \begin{bmatrix}x\\y\end{bmatrix}\right] \\ \text{s.t.} & Ax + By \leq b \\ & x \in \mathbb{R}^m \\ & y \in \mathbb{Z}^n \end{array} \tag{1.6}$$

   where $A \in \mathbb{R}^{k\times m}$, $B \in \mathbb{R}^{k\times n}$, $b \in \mathbb{R}^k$ and $Q_i \in \mathbb{R}^{(m+n)\times(m+n)}$ and $p_i \in \mathbb{R}^{m+n}$ for each $i \in \{1, \ldots, \ell\}$.

We define the following sets which we use alongside both MOMILP and MOMIQP:

1. $X := \{(x,y) \in \mathbb{R}^m \times \mathbb{Z}^n \colon Ax + By \leq b\}$
2. $\Psi := \{\psi \in \mathbb{R}^\ell : \psi = f(x,y) \text{ for } (x,y) \in X\}$

Notice that X, which we assume to be bounded, is the feasible set of (1.5) and (1.6), while Ψ is the collection of all points in $\mathbb{R}^\ell$ which can be formed using the objective function values of feasible solutions to either (1.5) or (1.6), whichever is appropriate.

When studying problems (1.5) and (1.6), one quickly finds that these problems are significantly challenging even in the presence of only two objectives, i.e., $\ell = 2$. As a result, in this work we consider the biobjective versions of (1.5) and (1.6), which we refer to as biobjective mixed-integer linear programs (BOMILP) and biobjective mixed-integer quadratic programs (BOMIQP), respectively. The next two sections focus, respectively, on BOMILP and BOMIQP. For each we discuss the state of the art in the literature, the places in which we found room for improvement, and provide a brief summary of our contributions.

1.3 Biobjective Mixed-integer Linear Programs

There are two main techniques found in the literature for solving BOMILPs with general integers. One is a variation of the branch-and-bound (BB) technique that is widely used for solving single-objective mixed-integer linear programs (MILP). Such a BB technique is proposed by Belotti et al. [10]. Note that all other BB techniques for BOMILP that we know of are designed for instances of BOMILP having only binary variables. The second method for solving BOMILPs is an iterative search method in the objective space, referred to as the "triangle-splitting" (TS) method, proposed by Boland et al. [16]. When employing TS, the objective space is iteratively partitioned into smaller and smaller search regions, each of which is either rectangular or triangular in shape, and various scalarization techniques for solving biobjective linear programs are used to search each region for solutions which are Pareto optimal. Additionally, Özpeynirci and Köksalan [68] give an exact method for finding supported solutions of BOMILP.

Other solution techniques in the literature are BB techniques that have been devoted to specific cases. Vincent et al. [101] improved upon the method of Mavrotas and Diakoulaki [63] for mixed 0-1 problems. Stidsen et al. [90] propose a method for solving mixed 0-1 problems in which only one of the objectives contains continuous variables. Kariwala and Cao [50] proposed methods for solving control structure design problems. The pure integer case has been studied for binary variables by Kiziltan and Yucaoğlu [53], general integers by Ralphs et al. [80] and specific classes of biobjective combinatorial problems by Jozefowiez et al. [48], Przybylski et al. [78] and Sourd and Spanjaard [85].

Although there is certainly merit in studying and developing objective space search methods for solving BOMILP, there is still much work that can be done to improve BB techniques for BOMILP and therefore we focus on BB methods for BOMILP in this work. In the following section we discuss the details of these techniques.

1.3.1 Overview of State-of-the-Art Methodology

In this section we discuss the methodology available in the literature for solving BOMILP using BB based procedures. We begin with a brief review of traditional BB methods.

Most modern BB procedures consist of two phases, which we refer to as the *Initialization Phase* and the *Main Step.* Given an instance of MILP (or BOMILP), the Initialization Phase is used to accomplish two tasks prior to beginning the Main Step: (i) reduce the size and complexity of a given instance, and (ii) determine an initial set of integer feasible solutions. It has been shown in the literature that both of these procedures have significant impact on the overall performance of BB. The Main Step of BB is then an iterative procedure consisting of two key procedures: (i) node processing, and (ii) branching. We describe these procedures in more detail during the following discussion.

During an iteration of a typical BB procedure for MILP (and by extension for BOMILP), one solves an LP subproblem at a node η selected from a list $\mathscr{L}$ of open subproblems in the BB enumeration tree. Then if some y_i takes a fractional value γ_i in the LP optimal solution, a process referred to as *branching* is performed. During branching two new subproblems are created by adding the constraints $y_i \geq \lceil \gamma_i \rceil$ and $y_i \leq \lfloor \gamma_i \rfloor$, respectively, to the LP subproblem at η. These two subproblems are then added to $\mathscr{L}$ and a new iteration is begun by selecting a new node $\eta' \in \mathscr{L}$ to explore and subdivide, if necessary. By continually selecting subproblems to explore and subdivide, a BB tree of subproblems is formed. In the single objective case, a fractional solution to the LP at node η provides a valid lower bound for all subproblems of η, while the objective function value associated with any integer feasible solution is an upper bound for every subproblem of the BB tree. In general, BB methods are effective because by comparing this global upper bound with the lower bound of a particular subtree, one is often able to prove that the subtree cannot provide a better integer feasible solution. Once this has been done, it is said that this subtree has been *pruned* or *fathomed.* Thus, in most cases the entire BB tree does not need to be explored and the best integer feasible solution can be found relatively quickly.

In the biobjective case, the bounds used for fathoming are no longer singletons in $\mathbb{R}$. Instead, they are subsets of $\mathbb{R}^2$ formed by taking unions of finitely many continuous convex piecewise linear functions [28]. For this reason, we refer to these bounds as *bound sets* throughout the remainder of this work. During each iteration s of BB, a node η_s of the BB tree is considered. The set of Pareto solutions to the LP relaxation of (2.1) associated with η_s provides a lower bound set $\mathcal{L}_s$. The upper bound set $\mathcal{U}_G$, on the other hand, is globally valid to all nodes of the BB tree, although it is generally not known in its entirety until completion of BB. Therefore, at each iteration s of BB, since $\mathcal{U}_G$ is unknown and cannot be used for fathoming, the set $\mathcal{U}_s = \vartheta(\mathcal{N}_s)$ is used instead. Here $\mathcal{N}_s \subset \Psi$ is the nondominated subset of the set of integer feasible solutions discovered prior to iteration s of BB and $\vartheta(\cdot)$ is a mapping which we describe in more detail after presenting a few new definitions. For use in these definitions, let S be a subset of $\mathbb{R}^2$.

Definition 1.8. A point $(\kappa_1, \kappa_2) := \kappa \in S$ is said to be *isolated* in S if there exists $\epsilon > 0$ for which $B_\epsilon(\kappa) := \{\hat{\kappa} \in S \setminus \{\kappa\} : \|\kappa - \hat{\kappa}\|_2 < \epsilon\}$ is empty.

Definition 1.9. Given distinct $\overline{\kappa}, \kappa' \in S$ such that $\overline{\kappa}_1 < \kappa'_1$ and $\overline{\kappa}_2 > \kappa'_2$, the point $\kappa^n = (\kappa'_1, \overline{\kappa}_2)$ is called the *local nadir point* with respect to $\overline{\kappa}$ and κ'.

Definition 1.10. Any line segment containing points in S is referred to as a *local nadir set.*

We now describe the mapping $\vartheta(\cdot)$, which can be used to construct $\mathcal{U}_s$ given $\mathcal{N}_s$. Notice that if a line segment contains no dominated points, then it must have a negative slope. We use the notation $[\kappa^{nw}, \kappa^{se}]$ to denote any such segment, where $\kappa^{nw} := (\kappa_1^{nw}, \kappa_2^{nw})$ and $\kappa^{se} := (\kappa_1^{se}, \kappa_2^{se})$ are the segment's north-west and south-east endpoints, respectively. Now, since at any iteration s of BB $\mathcal{N}_s \subset \Psi$ contains no dominated points, each of its elements must be either an isolated point or a line segment with a negative slope. For each point $\kappa = (\kappa_1, \kappa_2) \in \mathcal{N}_s$ consider κ_1, and for each segment $[\kappa^{nw}, \kappa^{se}] \in \mathcal{N}$ consider κ_1^{nw}. Arrange the elements of $\mathcal{N}_s$ in increasing order of these values. Then for each pair of adjacent elements $(\varepsilon^1, \varepsilon^2) \in \mathcal{N}_s$, if the south-east-most point of ε^1 is not equal to the north-west-most point of ε^2, calculate the local nadir point with respect to these two points and add it to a set $\mathcal{N}'_s$. Note that if ε^i for $i \in \{1, 2\}$ is a point and not a segment, then its north-west-most and south-west-most points are simply ε^i itself. Now let $\mathcal{N}''_s$ be the set of local nadir sets in $\mathcal{N}_s$. Then $\vartheta(\mathcal{N}_s) := \mathcal{N}'_s \cup \mathcal{N}''_s$ and thus $\mathcal{U}_s = \mathcal{N}'_s \cup \mathcal{N}''_s$. Figure 1.1 illustrates the relationship between $\mathcal{N}_s$ and $\mathcal{U}_s$.

One of the fathoming rules presented by Belotti et al. [10] states that at iteration s of BB a node η_s can be fathomed if $\mathcal{L}_s$ is separable from $\mathcal{U}_s$, i.e., $\mathcal{L}_s \cap (\mathcal{U}_s - \mathbb{R}^2_+) = \emptyset$. This is essentially the extension of the well known "fathoming by bound dominance" rule for single-objective problems to the biobjective case. Figure 1.2 shows examples of lower bound sets $\mathcal{L}_{s_1}$ and $\mathcal{L}_{s_2}$. Notice that the locations of these sets show that node η_{s_1} cannot be fathomed but η_{s_2} can. Clearly, efficient

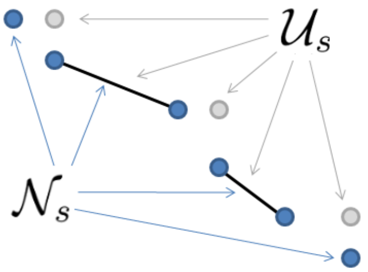


Figure 1.1: Examples of $\mathcal{N}_s$ and $\mathcal{U}_s$

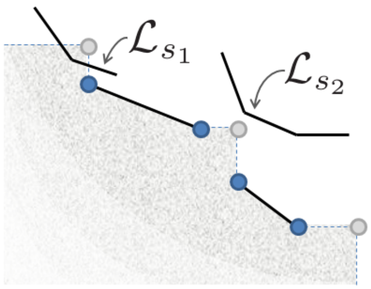


Figure 1.2: $\mathcal{L}_{s_1}$ is not separable from $\mathcal{U}_s$, but $\mathcal{L}_{s_2}$ is.

fathoming depends on the choice of $\mathcal{N}_s$ used to construct $\mathcal{U}_s$ since good approximations of $\mathcal{U}_G$ at each iteration of BB can help to fathom a large number of nodes.

Recognize that calculating $\mathcal{N}_s$ at iteration of BB can be quite cumbersome. Until now, two approaches have been used for this purpose. In the first approach, each time a point or segment $\psi \in \Psi$ is found is is stored in a list. Then dominated points and segments are removed from the list by performing a pairwise comparison between all stored solutions. After completion of the pairwise comparison, the stored data is precisely $\mathcal{N}_s$, and $\mathcal{U}_s$ can be constructed as $\vartheta(\mathcal{N}_s)$ [63, 101]. The second approach avoids updating throughout the BB procedure. Instead, a preprocessing phase is used to create a set $\mathcal{N} \subset \Psi_P$ before beginning BB. Then at every iteration s of BB, $\mathcal{N}$ is used in place of $\mathcal{N}_s$. Therefore a single set $\mathcal{U} = \vartheta(\mathcal{N})$ is used for fathoming throughout the BB [10].

1.3.2 Scope for Improvement

When studying BB techniques for BOMILP, we first discovered that there is still a significant need for the development and implementation of a BB procedure for instances of BOMILP containing general integer variables. We found, however, that in order to develop and implement a BB procedure capable a performing comparably with the state of the art Triangle Splitting method, improvements were needed in the following areas.

1. Methods for storing and dynamically updating $\mathcal{N}_s$: As discussed in the previous section, at iteration s of BB populating $\mathcal{N}_s$ is a non-trivial task, and both of the mentioned approaches lack each in their own way. Although storing each $\psi \in \Psi$ as it is discovered is beneficial, updating the list of these points via pairwise comparison is computationally expensive. On the other hand, pre-populating a set $\mathcal{N}$ with an appropriately chosen number of points from

Ψ and using $\mathcal{N}$ for fathoming at every iteration of BB can avoid this computational expense, but the strength of fathoming is sacrificed.

2. Procedures for presolving an instance of BOMILP: For single objective BB it is well known that performing sophisticated techniques to reduce the size of an instance of MILP or generate a "strong" set of feasible solutions for an instance of MILP prior to beginning the Main Step of BB can significantly reduce the computational effort needed to solve the instance. To date, there has been no work done to test the effectiveness of similar techniques in the biobjective setting.

We also point out that while studying solution methods for BOMILP we discovered that current methods have only been applied to relatively small instances of BOMILP, all of which have been randomly generated. Hence, there is still a need to test these techniques on larger, more realistic instances.

1.3.3 Contribution

We present a new data structure in the form of a modified binary tree that can be used to efficiently store, update, search and return the set of currently known nondominated solutions at any time during a BB procedure for solving BOMILP. This structure takes points and line segments in $\mathbb{R}^2$ as input and stores the nondominated subset of this input. We note that although the primary motivation for developing this structure was for its usefulness alongside BB algorithms, it is quite likely that the structure is also very useful when used alongside objective space search methods for solving BOMILP, such as the triangle splitting method, and also heuristics designed for approximating the Pareto set of BOMILP. We also note that when used alongside any exact solution procedure, such as a BB algorithm, at termination the data stored in the structure is precisely the set of Pareto optimal solutions. We perform three computational experiments. The first is designed to compare the utility of our structure for storing nondominated data to that of a dynamic list which updates via pairwise comparison. The results of our first experiment suggest that the data structure performs reasonably well in handling input of up to 10^7 points or segments and is much more efficient than a dynamic list. In the second and third experiments, we use our data structure alongside the biobjective BB techniques available in the literature and solve specific instances of BOMILP. We observe that the BB running times improve significantly when our structure is utilized.

Additionally, we present a generic BB method for finding all the Pareto solutions for BOMILP. We provide new algorithms for obtaining dual bounds at a node, for checking node fathoming, presolve and duality gap measurement. Our various procedures are implemented and empirically validated on instances from the literature. The results show that our BB technique performs extremely well, even better that the current state-of-the-art triangle splitting method. We then propose a new set of hard instances which we obtain by adding secondary objectives to single objective MIP instance from the MIPlib 2010 library. The majority of the problems in the MIPlib 2010 library come from real world problems and are considered to be challenging. We perform a final set of computational experiments using these instances. The results show that even for large, challenging instances our BB algorithm is competitive with the state-of-the-art triangle splitting method.

1.4 Biobjective Mixed-Integer Quadratic Programs

Currently there exist no techniques for solving BOMIQPs. One way in which such a technique can be constructed is by extending each of the key aspects of the BB technique we present for BOMILP to the context of BOMIQP. However, there is one major tool which is available for BOMILP that plays a crucial role in BB techniques for BOMILP, but which currently does not have a suitable counterpart available for BOMIQP. This tool is an algorithm known as the Parametric Simplex Algorithm (PSA).

The PSA is a parametric algorithm which uses a modified form of the weighted sum problem given in (1.4) and the result of Proposition 1.6 to find the set of all Pareto optimal solutions to the BOLP. The specific problem considered by the PSA is as follows:

$$\begin{array}{ll} \min\limits_{x} & \tilde{h}(x) := \lambda g_1(x) + (1-\lambda) g_2(x) \\ \text{s.t.} & \qquad\qquad (x) \in \mathcal{X} \end{array} \tag{1.7}$$

Recognize that solutions to (1.7) satisfy the conditions of Proposition 1.6 whenever $\lambda \in (0,1)$. The PSA begins by solving (1.7) for $\lambda = 1$ and then uses the reduced costs of each nonbasic variable with respect to each of the objective functions in order to determine the range of values $\lambda \in [\lambda', 1]$ for which the current basis is optimal. The algorithm then returns a nonbasic variable and a basic variable which need to enter and leave the basis, respectively, in order to create an alternative basis that will be optimal for $\lambda = \lambda'$. The algorithm then repeats iteratively until an optimal basis is discovered for all $\lambda \in (0, 1)$.

Hirschberger et al. [44], Romanko [82] and Steuer et al. [89] present methods which perform similarly to the PSA but are designed for the subclass of BOQP in which one objective contains only linear terms. Goh and Yang [36] present an algorithm for general BOQP that performs similarly to the PSA, but it requires the computation of several matrix inverses and depends on other calculations that are unnecessarily ambiguous. Arora et al. [4], Ehrgott et al. [29], Goh and Yang [37], Ohsawa [66], Peng et al. [73] provide algorithms for specialized subclasses of BOQP. Beato-Moreno et al. [9] give a technique for solving unconstrained BOQP with strictly convex objective functions.

In order to develop an improved analogue for the PSA for general BOQP, we first develop the theory and methodology necessary to describe the properties of the solution sets of parametric and multiparametric single objective QPs. Recognize that due to the result of Proposition 1.6, developing a tool for solving multiparametric QP is enough in order to establish a counterpart to the PSA for BOQP. Due to the complicated nature of general multiparametric QP, our analysis is divided into two sections. The first covers the cases in which parameters are present in the linear terms of the objective function as well as the right hand sides of the constraints. The second extends the concepts developed in the previously mentioned restricted case to the general case in which parameters are permitted in any location within the QP.

1.4.1 Overview of State-of-the-Art Methodology for Parametric QP

Single objective QPs containing a single parameter are referred to as parametric QPs (pQP). These problems have been widely studied. In fact, one of the earliest studies of pQP was conducted in 1962 and is due to Ritter [81]. More recently, QPs containing two parameters, referred to as biparametric QP (bpQP), and more than two parameters, referred to as multiparametric QP (mpQP), have been studied.

There are a variety of goals that one may have when solving a parametric program. Ghaffari-Hadigheh et al. [34] categorize these goals into three types of sensitivity analysis, which they refer to as Type I, Type II, and Type III. Suppose that $\theta \in \mathbb{R}^d$ is the vector of parameters associated with a given parametric or multiparametric QP, denoted $QP(\theta)$. It is assumed that $\theta \in S_\theta \subset \mathbb{R}^d$, where S_θ is a convex polyhedron. The goal of each type of sensitivity analysis is then to partition S_θ into regions with the following properties:

- Type I - For all θ within a given region, $QP(\theta)$ has a solution with the same optimal basis.
- Type II - For all θ within a given region, $QP(\theta)$ has an optimal solution with the same set of active constraints.
- Type III - For all θ within a given region, $QP(\theta)$ has an optimal solution with the same representation of the primal and dual variables as functions of θ.

In the literature these regions have been given a variety of names, such as invariancy regions, critical regions, and validity sets. We will refer to them as invariancy regions. Also, note that Type I sensitivity analysis performs poorly in the presence of degeneracy and multiple optima and therefore has not been widely used for parametric QP.

Ghaffari-Hadigheh et al. [34, 35] and Romanko [82] perform Type III analyses on bpQPs in which one parameter is contained in the linear term of the objective function, and the other is in the right hand side of the constraints. In these methods S_θ is split into an explored region and an unexplored region. An interior point is calculated within the unexplored region, and then a single-parametric technique is used to find all edges and vertices bounding the invariancy region containing the generated interior point. This process is then repeated until the unexplored region is empty.

Baotić [5], Bemporad et al. [12] and Tøndel et al. [94] perform Type II analyses on mpQPs in which all parameters are in the right hand side of the constraints. These methods split S_θ into an explored region and one or more unexplored regions. Interior points are calculated for each unexplored region and then the set of active constraints at the optimal solution associated with each interior point is used to determine a polyhedral representation of the invariancy region containing the interior point. This process is then repeated until all unexplored regions are empty. Note that the polyhedral representations of the invariancy regions often contain many redundant and therefore unnecessary inequalities. It is mentioned that these redundant inequalities should be discarded, but no discussion is given as to the process of classifying each inequality as redundant or not. Also note that the methods of Baotić [5] and Bemporad et al. [12] require that the set of active constraints associated with every optimal solution be linearly independent. Tøndel et al. [94] discuss a method for selecting the interior points of each unexplored region in such a way that the linear independence requirement is not needed.

Spjøtvold et al. [87] and Tøndel et al. [95] perform type II analyses on mpQP containing parameters in both the linear term of the objective as well as the right hand side of the constraints. These methods are very similar to that of Tøndel et al. [94]. Gupta et al. [40] also consider these problems, but they use an enumeration technique to explore all possible sets of active constraints. They also employ "pruning" techniques to show that certain sets of active constraints will not provide any new invariancy regions. We note that Grancharova and Johansen [39] provide an extensive summary of the works pertaining to mpQP.

It is common for researchers studying mathematical optimization to consider problems of interest in a variety of equivalent formulations. Hence, as we continue our study of QP and mpQP, it is important to note that there exists a problem, known as the *Linear Complementarity Problem* (LCP), which arises naturally when studying duality theory in the context of QP. Furthermore, under certain assumptions, it can be shown that LCP and QP are equivalent. Consider the following definition and two related propositions.

Definition 1.11. Given $M \in \mathbb{R}^{n \times n}$ and $q \in \mathbb{R}^n$, the *Linear Complementarity Problem (LCP)* is the problem of finding vectors $w, z \in \mathbb{R}^n$ which satisfy:

$$\begin{gathered} w - Mz = q \\ w^\top z = 0 \\ w, z \geq 0 \end{gathered} \tag{1.8}$$

We now provide two definitions which play a crucial role throughout this work. For use in these definitions we denote the LCP in (1.8) as LCP(q, M).

Definition 1.12. A matrix $M \in \mathbb{R}^{n \times n}$ is *column sufficient* if the following implication is satisfied:

$$(x_i(Mx)_i \leq 0 \text{ for all } i) \quad \Rightarrow \quad (x_i(Mx)_i = 0 \text{ for all } i) \tag{1.9}$$

M is said to be *row sufficient* if $M^\top$ is column sufficient. If M is both column and row sufficient, it is then called *sufficient*.

Definition 1.13. A matrix $M \in \mathbb{R}^{n \times n}$ is said to be a Q_0 matrix if the set

$$\mathcal{K}(M) = \{q \in \mathbb{R}^n : \text{LCP}(q, M) \text{ has at least one solution}\}$$

is a convex cone.

Proposition 1.14. Every QP can be reformulated as an LCP.

Proof. Consider a general QP of the form

$$\begin{array}{lll} \min\limits_{x} & f(x) = \dfrac{1}{2}x^\top Qx + p^\top x & \\ \text{s.t.} & Ax \leq b & \\ & x \in \mathbb{R}^t & \end{array} \tag{1.10}$$

Assume without loss of generality that the variables of (1.10) are restricted to be nonnegative (if not, this property can be achieved using a simple reformulation). Convert all inequality constraints of (1.10) to equality constraints by adding slack variables s. It is well known that a necessary condition for the optimality of a feasible solution $(\hat{x}, \hat{s})$ to this problem is that there exist Lagrange multipliers, or dual variables, $\hat{u}$ and $\hat{r}$, associated with the constraints $Ax \leq b$ and $x \geq 0$, respectively, which satisfy the following Karush-Kuhn-Tucker First Order Necessary Conditions (KKT FONCs):

$$\begin{array}{l} -Q\hat{x} - A^\top \hat{u} + \hat{r} = p \\ A\hat{x} + \hat{s} = b \\ \hat{u}^\top \hat{s} = 0 \\ \hat{r}^\top \hat{x} = 0 \\ \hat{x}, \hat{s}, \hat{u}, \hat{r} \geq 0 \end{array} \tag{1.11}$$

Let $M = \begin{bmatrix} 0 & -A \\ A^\top & Q \end{bmatrix}$, $q = \begin{bmatrix} b \\ p \end{bmatrix}$, $w = \begin{bmatrix} \hat{s} \\ \hat{r} \end{bmatrix}$ and $z = \begin{bmatrix} \hat{u} \\ \hat{x} \end{bmatrix}$ and notice that substituting into (1.11) gives a system identical to (1.8). □

Proposition 1.15. Consider a QP as in (1.10). If Q is positive semi-definite (PSD) then a solution to the LCP reformulation of the QP is guaranteed to provide an optimal solution to the QP.

Proof. It is well known that if Q is PSD, the KKT FONCs are sufficient for optimality. The result follows. □

It is also important to note that when an instance of QP for which Q is PSD is reformulated to obtain LCP(q, M), the matrix M will also be PSD. Furthermore, since the class of PSD matrices is a subset of the class of sufficient matrices, M is also sufficient.

The LCP is a well known problem in the literature and has been studied extensively by researchers such as Cottle et al. [22], Kostreva [55], Lemke [56] and Murty and Yu [65]. The results of Proposition 1.14 and 1.15 are also well known and their proofs can be found, for example, in [65].

Due to the results of Propositions 1.14 and 1.15, it is also necessary to mention works which consider parametric LCP. As with single parametric QP, single parametric LCP (pLCP) has been studied for quite some time. One of the earlies works was published in 1971 and is due to Murty [64]. Multiparametric LCP (mpLCP), however, has been studied more recently. Note that in solving the LCP associated with a given QP, one will find optimal values for both the primal and dual variables of the QP. Thus, by performing a Type I sensitivity analysis on an LCP (the idea of a basis for an LCP will be introduced later), one is performing a Type III sensitivity analysis on the associated QP. Columbano et al. [18], Gailly et al. [32], Herceg et al. [42], Jones and Morari [46] and Li and Ierapetritou [57] consider mpLCP with parameters in the vector q. The method of Gailly et al. [32] is theoretically sound but it lacks any practical discussion as to how the theory should be implemented. Although the authors of [18, 42, 46, 57] do provide practical details as to how to implement their methods, there is still significant room for improvement in each method. The techniques of Columbano et al. [18] and Jones and Morari [46] work well in the absence of degeneracy, but when degeneracy is present they depend on an ϵ-perturbation of the vector q and require postprocessing in order to obtain the solution for the original problem from its ϵ-perturbed counterpart. The method of Li and Ierapetritou [57] is quite computationally costly since it requires reformulating the mpLCP as a multiparametric bilinear mixed integer program. Herceg et al. [42] propose an enumerative approach which extends the method of Gupta et al. [40] for mpQP to the context of mpLCP.

Chakraborty et al. [17], Tammer [93], Väliaho [98] and Xiao [103] consider mpLCP with parameters in M. However, the works of Tammer [93] and Xiao [103] do not discuss finding invariancy regions, but instead discuss interesting properties of the solutions under various assumptions. Although the works of Chakraborty et al. [17] and Väliaho [98] do present methodology for solving parametric LCP, the work of Chakraborty et al. [17] imposes very strict limitations on the structure of the matrix M and the work of Väliaho [98] is only applicable for the single parametric case, i.e., $k = 1$.

1.4.2 Scope for Improvement

For QP, the case in which a single parameter is present either in the linear term of the objective or the right hand side of the constraints has been studied extensively, as discussed in the previous section. Thus, we focus on cases in which at least one of the following conditions is met:

(i) one or more parameters exist in the quadratic term of the objective, (ii) at least two parameters exist in the linear term of the objective and/or the right hand side of the constraints. For LCP, the case in which a single parameter is present in q has been studied extensively, as also discussed in the previous section. Thus, we focus on cases in which at least one of the following conditions is met: (i) one or more parameters exist in M, (ii) at least two parameters exist in q. We note that mpQP with two or more parameters in Q and mpLCP with two or more parameters in M have not yet been solved in the literature.

Recall from the previous subsection that most of the methods for solving mpQPs depend on the computation of an interior point which is used to determine a representation for each invariancy region. As it is difficult to ensure that a point is selected which falls outside of all previously explored invariancy regions, methods depending on such a calculation are often inefficient. The methods for mpLCP, however, do not have this issue. Currently the best methods for solving mpLCP are those of Columbano et al. [18] and Jones and Morari [46]. As mentioned before, though, these techniques lack a straightforward implementation in the case of degeneracy.

We also point out the following two very important questions that are yet unanswered in the areas of mpQP and mpLCP:

1. Given an instance of mpQP or mpLCP, how does one determine an initial full dimensional invariancy region?

2. How does one solve an instance of mpQP with parameters in the quadratic term of the objective, or mpLCP with parameters in M?

1.4.3 Contribution

We present two new algorithms, each consisting of two phases, designed for solving mpLCP. The first of these algorithms is designed for the case in which the matrix M is a real-valued sufficient matrix (we define sufficiency shortly). This algorithm improves on the methods of Jones and Morari [46] and Columbano et al. [18], but never relies on an ϵ-perturbation, functioning the same whether or not degeneracy is present. Furthermore, the worst-case complexity of the algorithm matches that of Columbano et al. [18] for nondegenerate problems and is lower than that Columbano et al. [18] for degenerate problems. A computational study also shows that the proposed algorithm significantly outperforms the current state of the art implementation for solving mpLCP, the Multi-Parametric

Toolbox [41] available for MATLAB, which implements the algorithms presented in Jones and Morari [46].

The second algorithm we present is an extension of the first, and is designed for solving mpLCP when parameters are present in both the vector q and the matrix M, which we assume to be sufficient at every "attainable" value of the parameters. The reason we present these algorithms separately is that the theoretical arguments needed in order to prove the correctness of the second algorithm are significantly more challenging. In fact, these arguments require a thorough knowledge of many concepts from algebraic geometry and algebraic topology in addition to mathematical optimization and operations research.

Role of Sufficient Matrices in mpLCP: We now discuss the important role that sufficiency plays throughout this work. The following important property of column sufficient matrices is shown in [22]: If M is column sufficient, the set $\mathcal{K}(M)$ is partitioned by a set of complementary cones (both Chapters 4 and 5 contain explicit definitions of complementary cones). Together, this property of column sufficient matrices and the property of Q_0 matrices given in Definition 1.13 ensure that the set of "attainable" parameters for mpLCP can be partitioned into a set of *invariancy regions* such that the representation of the mpLCP solution vectors w and z as functions of the parameters is invariant over each region (more details and a justification of this statement are provided in Chapters 4 and 5). As a result, the algorithms we propose in Chapters 4 and 5 for solving mpLCP are designed for instances of mpLCP in which M is both Q_0 and column sufficient. As the largest class of matrices known to be a subset of both the classes of Q_0 and column sufficient matrices is the class of sufficient matrices (see [21]), we therefore assume that M is sufficient.

It is also important to note that in the first phase of each of the proposed algorithms, we provide an answer to the following previously unanswered question: Given an instance of mpLCP, how does one determine an initial full dimensional invariancy region?

1.5 Organization of the Dissertation

The remaining chapters of this dissertation, with the exception of Chapter 6, are slightly modified versions of individual papers that have been, or will soon be, submitted to various optimization and operations research journals. The journal of submission will be indicated at the start of each chapter. Note that due tothe independence of each of these chapters, appropriate notation and definitions will be presented within each, and may occasionally overlap.

In Chapter 2 we present the data structure discussed in Section 1.3.3. We provide all details necessary for the implementation of this data structure and present the results of various computational experiments showing the utility of the structure.

Chapter 3 contains a proposed BB algorithm for solving BOMILP. We inspect each of the key aspects of traditional BB algorithms for single objective mixed integer programs, and for each develop an analogue for BOMILP. We then perform a variety of computational tests, each of which highlights an important aspect of our algorithm. Finally, we propose a new set of challenging instances of BOMILP and test the performance of our algorithm on this set.

Chapters 4 and 5 contain the algorithms we propose for solving mpLCP with (i) parameters only in the vector q, and (ii) parameters present both in q and the matrix M, respectively. Each chapter contains a significant amount of theory which serves as the necessary foundation for the development of the presented algorithms. In Chapter 4 we provide the results of a computational test which shows that our algorithm convincingly outperforms other current methods for solving mpLCP with parameters in q. A similar comparison cannot be given in Chapter 5, however, because the problem we consider had previously been unsolved.

We conclude the dissertation with Chapter 6 in which we discuss possible directions for future research.

Chapter 2

Efficient storage of Pareto points in BOMILP

[The contents of this chapter include material from a paper entitled "Efficient storage of Pareto points in biobjective mixed integer programming," which was submitted to the *INFORMS Journal on Computing* in September of 2014; the authors are N. Adelgren, P. Belotti and A. Gupte. The paper is currently undergoing a second round of review.]

2.1 Introduction

Biobjective mixed integer linear programs (BOMILP) have the following form,

$$\begin{array}{cl} \min_{x,y} & f(x,y) := \left[f_1(x,y) := c_1^\top x + d_1^\top y,\ f_2(x,y) := c_2^\top x + d_2^\top y\right] \\ \text{s.t.} & (x,y) \in P_I := \{(x,y) \in \mathbb{R}^m \times \mathbb{Z}^n \colon Ax + By \leq b\} \end{array} \tag{2.1}$$

where P_I is a bounded set. Thus BOMILP encompasses both biobjective linear programs (BOLP) and biobjective integer programs (BOIP). Define $\Psi := \{\psi \in \mathbb{R}^2 : \psi = f(x,y)\ \forall (x,y) \in P_I\}$ to be the collection of all points in $\mathbb{R}^2$ that can be obtained using the objective function values of feasible solutions to (2.1). We refer to the space $\mathbb{R}^2$ containing Ψ as the *objective space.*

Unlike single-objective programs, one cannot expect to find a single optimal solution to biobjective programs since the objective functions are oftentimes conflicting. Instead, a set of efficient solutions which offer an acceptable compromise between the objectives is sought. In order to determine what types of solutions are "acceptable," we provide several notations and definitions. For any two vectors $v^1, v^2 \in \mathbb{R}^2$ we use the following notation: $v^1 \leqq v^2$ if $v^1_i \leq v^2_i$ for $i = 1, 2$; $v^1 \leq v^2$ if $v^1 \leqq v^2$ and $v^1 \neq v^2$; and $v^1 < v^2$ if $v^1_i < v^2_i$ for $i = 1, 2$. Given distinct $(\overline{x}, \overline{y}), (x', y') \in P_I$, we say that $f(\overline{x}, \overline{y})$ *dominates* $f(x', y')$ if $f(\overline{x}, \overline{y}) \leq f(x', y')$. This dominance is *strong* if $f(\overline{x}, \overline{y}) < f(x', y')$; otherwise it is *weak*. A point $(\overline{x}, \overline{y}) \in P_I$ is (*weakly*) *efficient* if $\nexists\ (x', y') \in P_I$ such that $f(x', y')$ (strongly) dominates $f(\overline{x}, \overline{y})$. The set of all efficient solutions in P_I is denoted by X_E. A point $\overline{\psi} = f(\overline{x}, \overline{y})$ is called *Pareto optimal* if $(\overline{x}, \overline{y}) \in X_E$. Given $\Psi' \subseteq \Psi$ we say that $\psi' \in \Psi'$ is *nondominated* in Ψ' if $\nexists\ \psi'' \in \Psi'$ such that ψ'' dominates ψ'. Note that Pareto optimal points are nondominated in P_I. A BOMILP is considered solved when the set of Pareto optimal points $\Psi_P := \{\psi \in \mathbb{R}^2 : \psi = f(x, y)\ \forall (x, y) \in X_E\}$ is found. Note that the definitions given here extend to problems with more than two objectives, but we give them for biobjective problems since that is our focus in this chapter.

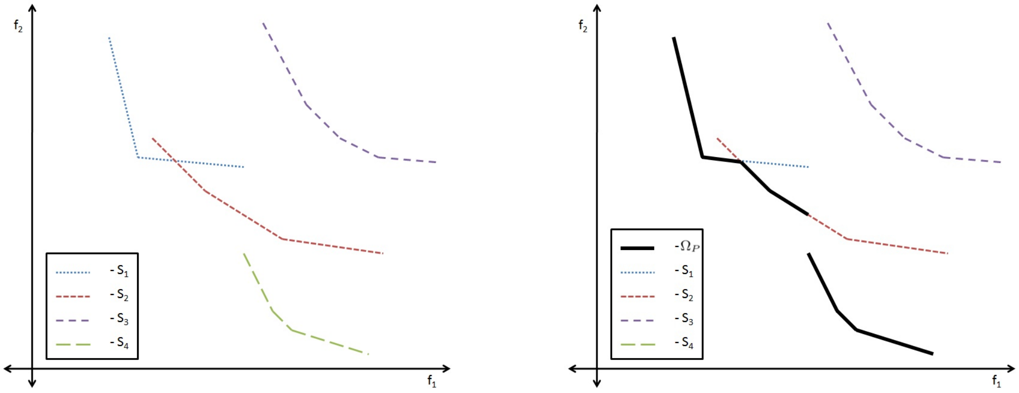
It is well-known [cf. 27] that a BOLP can be solved by taking convex combinations of $f_1(\cdot)$ and $f_2(\cdot)$ and solving a finite number of LPs. Thus for BOLP, the set of Pareto points can be characterized as $\Omega_P = \{(\xi_1, \xi_2) \in \mathbb{R}^2 \colon \xi_2 = \psi(\xi_1)\}$ where $\psi(\cdot)$ is a continuous convex piecewise linear function obtained using extreme points of the dual feasible region. Similarly, for BOIP it is known that Ω_P is a finite set of discrete points in $\mathbb{R}^2$. Now consider the case of BOMILP. Let $Y = \operatorname{Proj}_y P_I$ be the set of integer feasible subvectors to (2.1). Since P_I is bounded, we have $Y = \{y^1, \dots, y^k\}$ for some finite k. Then for each $y^i \in Y$ there is an associated BOLP, referred to as a *slice problem* and denoted $\mathbb{P}(y^i)$, obtained by fixing $y = y^i$ in (2.1),

$$\mathbb{P}(y^i) \qquad \begin{array}{rl} \min_x & \{f_1(x) = c_1^\top x + d_1^\top y^i,\ f_2(x) = c_2^\top x + d_2^\top y^i\} \\ \text{s.t.} & Ax \leq b - By^i \end{array} \tag{2.2}$$

Problem $\mathbb{P}(y^i)$ has a set of Pareto solutions $S_i := \{(\xi_1, \xi_2) \in \mathbb{R}^2 \colon \xi_2 = \psi_i(\xi_1)\}$, where $\psi_i(\cdot)$ is a continuous convex piecewise linear function as explained before. Then $\Omega_P \subseteq \cup_{i=1}^k S_i$ and this inclusion is strict in general. In particular, we have:

$$\Omega_P = \overset{k}{\underset{i=1}{\cup}} \left(S_i \setminus \underset{j \neq i}{\cup} \left(S_j + \mathbb{R}^2_+ \setminus \{\mathbf{0}\} \right) \right) \tag{2.3}$$

Such a union of sets is not, in general, represented by a convex piecewise linear function. Figure 2.1 shows an example with $k = 4$.



(a) $\cup_{i=1}^{4} S_i$

(b) Ω_P

Figure 2.1: Relationship between Pareto sets of slice problems and original BOMILP

It should be noted that finding Ω_P is not a trivial task in general. In the worst case, $\Omega_P = \cup_{i=1}^{k} S_i$ and one may have to solve every slice problem to termination, which can have exponential complexity. For multiobjective IP's (i.e. $m = 0$), De Loera et al. [25] prove that Ω_P can be enumerated in polynomial-time for fixed n, which extends the well known result that single-objective IP's can be solved in polynomial-time for fixed n. We are unaware of any similar results for BOMILP.

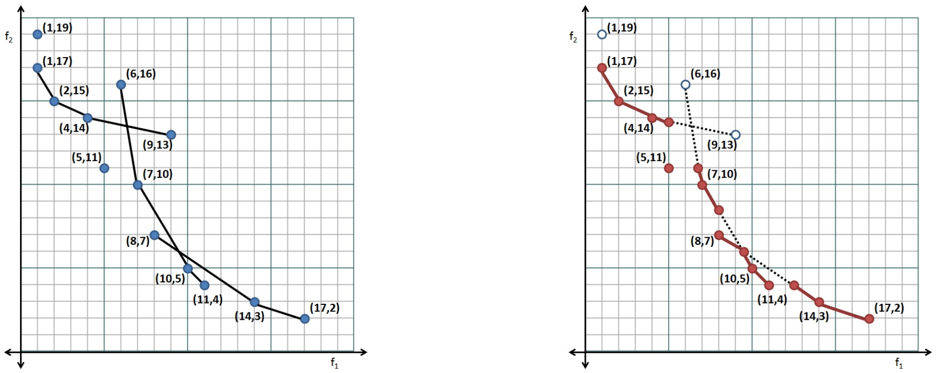
Not many exact procedures have been presented for solving BOMILP with general integers. The works of Belotti et al. [10] and Boland et al. [15] are the only ones we know of, though Özpeynirci and Köksalan [68] give an exact method for finding supported solutions of BOMILP. Most other techniques in the literature have been devoted to specific cases. Vincent et al. [101] improved upon the method of Mavrotas and Diakoulaki [63] for mixed 0-1 problems. Stidsen et al. [90] propose a method for solving mixed 0-1 problems in which only one of the objectives contains continuous variables. We point out that the works of Belotti et al. [10], Mavrotas and Diakoulaki [63], Stidsen et al. [90] and Vincent et al. [101] are based on biobjective branch-and-bound (BB) procedures in which the Pareto set is determined by solving several BOLPs, while the works of Boland et al. [15] an Özpeynirci and Köksalan [68] utilize other techniques in which the Pareto set is determined by solving several MIPs. We also note that the pure integer case has been studied for binary variables [53], general integers [80] and specific classes of biobjective combinatorial problems [48, 77, 85].

Given a set of feasible solutions to a BOMILP, the data structure we present in this chapter efficiently stores the nondominated subset of these feasible solutions. This structure is useful alongside exact solution procedures as well as heuristics aimed at approximating the Pareto set. Our structure is a modified version of a *quad-tree*. Although quad-trees (see, for example, Samet [83] for background and details) have been used extensively for storing Pareto points in the past [91, 92], they have been used only in the pure integer case. Sun and Steuer [92] stored nondominated solutions using both quad-trees and dynamic lists which were updated via pairwise comparison. They showed that for pure integer biobjective (and also multiobjective) problems, quad-trees were able to store nondominated solutions more efficiently than dynamic lists. In the pure integer case all nondominated solutions are singletons while in the mixed integer case nondominated solutions can consist of line segments as well as singletons. Our quad-tree is organized in such a way that it can be easily searched to find a desired subset of stored line segments and singletons; the need to do so arises, for example, when using branch-and-bound type methods to solve a BOMILP. The algorithms we use to implement this tree force it to remain balanced, which is significant because having a balanced tree reduces the time complexity required to access an individual node.

Section 2.2 describes our structure, provides the algorithms necessary for its implementation, and discusses the complexity and correctness of each of these algorithms. Section 2.3 provides an example of utilizing the structure to determine the nondominated subset of a particular set of solutions. In Section 2.4 we present the results of three experiments. The results of the first experiment show that in the mixed integer case our data structure is able to store nondominated solutions more efficiently than a dynamic list and, in most cases, can handle up to 10^7 inserted solutions in reasonable time. In the second and third experiments we utilize our structure alongside the BB procedures of Belotti et al. [10] and Adelgren and Gupte [1], respectively, to solve specific instances of BOMILP. The results show that the use of our structure leads to faster solution times for the majority of the solved instances of BOMILP.

2.2 Tree data structure

We begin this section by presenting the high-level idea of our data structure. Next we give a detailed description of the data structure and the algorithms we used to implement it. We finish by discussing some theoretical results including the complexity of each algorithm, and thus the overall structure. Throughout this discussion, when we refer to storing solutions we are referring to points in the objective space. Recall that we will be storing the nondominated subset of the union of several



(a) Generated points and segments

(b) Nondominated subset

Figure 2.2: Example of solutions generated when solving an instance of BOMILP.

Pareto sets. One convenient way to store this subset is to store each of the individual points and line segments in $\mathbb{R}^2$ that it comprises.

2.2.1 Purpose and principle

Figure 2.2a shows an example of solutions that might be generated when solving an instance of BOMILP. We would like to store the nondominated portion of these points and segments, as shown in Figure 2.2b. Our goal is to have a data structure $\mathcal{S}$ which can take points and line segments as input, and store only the nondominated subset of the solutions regardless of the order in which they are inserted. Therefore, when a new solution is added to $\mathcal{S}$, it needs to not only recognize whether or not the new solution is dominated by solutions already in $\mathcal{S}$, but it must also be able to determine whether or not the new solution dominates any currently stored solutions. Once these checks have been made, $\mathcal{S}$ must be able to update itself and store only nondominated solutions. Consider the set of solutions depicted in Figures 2.2a and 2.2b and suppose that the segments connecting (1,17), (2,15), (4,14), and (9,13) are currently stored in $\mathcal{S}$. When inserting the point (5,11) into $\mathcal{S}$, it must recognize that the point dominates a portion of the segment connecting (4,14) and (9,13), and thus this portion of the segment must be removed from $\mathcal{S}$ before the point is added to $\mathcal{S}$. Similarly, when the segment connecting (6,16) and (7,10) is inserted, it must recognize that a portion of this segment is dominated by (5,11) and therefore only allow the nondominated portion of the segment to be added. The data structure we use is a modified version of a quad-tree in which each node represents either a singleton or a line segment associated with a Pareto point or set of Pareto points of (2.1). Note that a quad-tree is a data structure specifically designed for storing data in $\mathbb{R}^p$, with

$p \in \mathbb{N}$. For the remainder of this work, though, we restrict our discussion of quad-trees to their application in $\mathbb{R}^2$. In this case, each node π in a quad-tree has at most four children, one for each quadrant of the Cartesian plane. The four children of π must lie within $\pi + \mathbb{R}_{++}, \pi + \mathbb{R}_{-+}, \pi + \mathbb{R}_{--}$ and $\pi + \mathbb{R}_{+-}$, respectively, where, for example, $\mathbb{R}_{++} := \{x \in \mathbb{R}^2 : x_1 \geq 0, x_2 \geq 0\}$.

2.2.2 Operations and details

Due to the fact that dominated solutions are not stored in our structure, our modified quad-tree actually reduces to a modified binary tree. Let Π be the set of nodes in the tree. For a given $\pi \in \Pi$, notice that if solutions are present in $\pi + \mathbb{R}_{++}$, they are dominated by π and should not be stored in the tree. Similarly, if solutions are present in $\pi + \mathbb{R}_{--}$, they dominate π and π should be removed from the tree. Thus, for any node $\hat{\pi} \in \Pi$ the children of $\hat{\pi}$ associated with $\hat{\pi} + \mathbb{R}_{++}$ and $\hat{\pi} + \mathbb{R}_{--}$ are unnecessary. Hence, each $\hat{\pi} \in \Pi$ has only two children, and thus the tree reduces to a binary tree.

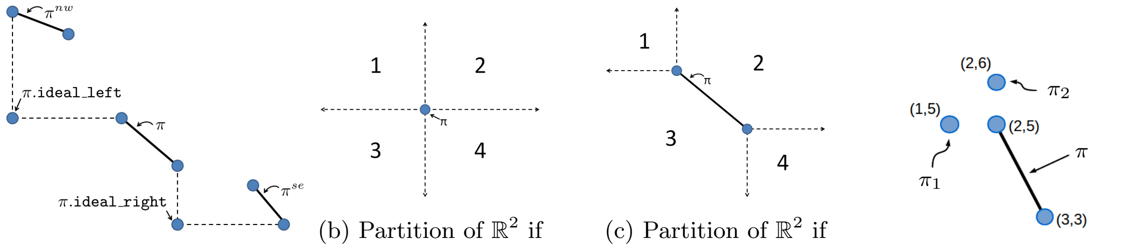
In order to present our structure in a clear, understandable manner, we define the following terms for each $\pi \in \Pi$:

1. $\pi.\texttt{type}$ - Sgmt for π representing line segment, and Pnt for π representing a singleton.
2. $\pi.x_1, \pi.x_2, \pi.y_1$ and $\pi.y_2$ - π is identified by:

   (i) Point $(\pi.x_1, \pi.y_1)$ if $\pi.\texttt{type} = \textsf{Pnt}$.

   (ii) Extreme points $(\pi.x_1, \pi.y_1)$ and $(\pi.x_2, \pi.y_2)$ if $\pi.\texttt{type} = \textsf{Sgmt}$.

   Note that if $\pi.\texttt{type} = \textsf{Pnt}$, we assume $(\pi.x_1, \pi.y_1) = (\pi.x_2, \pi.y_2)$.
3. $\pi.p$ - parent node of π.
4. $\pi.l$ - left child node of π.
5. $\pi.r$ - right child node of π.
6. $\pi.\texttt{size}$ = total # of nodes contained in the sub-tree rooted at π.
7. $\pi.\texttt{ideal\_left} = (\pi^{nw}.x_1, \pi.y_1)$ where π^{nw} is the north-west-most node in the subtree rooted at π.
8. $\pi.\texttt{ideal\_right} = (\pi.x_2, \pi^{se}.y_2)$ where π^{se} is the south-east-most node in the subtree rooted at π.



(a) Local ideal points.

(b) Partition of $\mathbb{R}^2$ if π.type = Pnt.

(c) Partition of $\mathbb{R}^2$ if π.type = Sgmt.

(d) Weak domination.

Figure 2.3: Visuals for π.ideal_right, π.ideal_left, the partitioning of $\mathbb{R}^2$ relative to π, and weak domination.

We say that π is the *root node* if $\pi.p = \emptyset$ and π is instead a *leaf node* if $\pi.l = \pi.r = \emptyset$. See Figure 2.3a for the details of π.ideal_left and π.ideal_right.

Now, in order to further simplify the descriptions of the algorithms we use in implementing our data structure, we partition $\mathbb{R}^2$ into 4 regions relative to any node π. Figures 2.3b and 2.3c show the details of this partition for each type of node. We denote these regions by $R_\alpha(\pi)$ where $\alpha \in \{1, 2, 3, 4\}$ represents the number of the region as shown in Figures 2.3b and 2.3c. Given distinct nodes π and π^*, we use the notation $\pi^* \cap R_\alpha(\pi)$ to denote any portion of the point or segment associated with node π^* that lies in region $R_\alpha(\pi)$. If no such portion exists, we say $\pi^* \cap R_\alpha(\pi) = \emptyset$. In order to ensure that these regions are disjoint let us assume that each region contains its lower and left boundaries, but not its upper or right boundaries. Also assume that π itself is contained in $R_2(\pi)$ and not $R_i(\pi)$ for $i \in \{1, 3, 4\}$. Note that this convention is taken so that weakly dominated points will not be included in our structure. This convention is convenient since we are working with minimization problems, but if working with maximization problems one should include upper and right boundaries in each region and include π in $R_3(\pi)$ rather than $R_2(\pi)$. Now, for an example, suppose $\pi \in \Pi$ is defined by the segment between (2,5) and (3,3). Further suppose that π_1 is the point (1,5), π_2 is the point (2,6) and both π_1 and π_2 are inserted into our structure. Observe Figure 2.3d. The point associated with node π_1 weakly dominates the left-most point of the segment associated with π and thus π_1 should be stored. However, the point associated with node π_2 is weakly dominated by the segment associated with π and so π_2 should not be stored. Situations like this motivate our decision to include lower and left boundaries with a given region, but not upper or right boundaries. Such a convention further simplifies the descriptions of the algorithms we use in implementing our data structure.

This data structure has three main purposes: (i) it should be able to handle the insertion of several thousands (or even millions) of points and segments and update itself efficiently, (ii) the structure must be organized so that it can easily be searched and a desired subset can be obtained, and (iii) it must be able to return the current set of nondominated solutions. So, the main algorithms needed for the utilization of this data structure are functions for *insertion* of new solutions, *deletion* of dominated solutions, and *rebalancing* of the tree. We describe these algorithms next.

2.2.2.1 Insertion.

Recall that $\Omega_P \subseteq \bigcup_{i=1}^{k} S_i$ and is hence a collection of points and segments. Thus only points or segments will be inserted into the structure. For this purpose we define the Insert function which takes two inputs: a node π^* which is being inserted and a node π which is the root of the tree or subtree where π^* is inserted. The point or segment associated with π^* is compared against π. Consider the following four situations:

1. If $\pi^* \subseteq R_2(\pi)$ then $\pi \leqq \pi^*$ and thus π^* is discarded.

2. If $\pi \nleqq \pi^*$ but $\pi^* \cap R_2(\pi) \neq \emptyset$ then a portion of π^* is either dominated by π or is a repetition of a solution stored in π. We denote this situation by $\pi \leqq_p \pi^*$. In this case $\pi^* \cap R_2(\pi)$ is discarded.

3. If $\pi^* \leqq \pi$ then π is removed from the tree.

4. If $\pi^* \nleqq \pi$ but $\pi \cap R_2(\pi^*) \neq \emptyset$ then π is reduced to $\pi \setminus R_2(\pi^*)$.

Note that the second possibility above may result in π^* being split into two pieces. Similarly, the final possibility may result in π being split into two nodes. If none of the above four scenarios occur, then neither π nor π^* dominates the other and they can therefore coexist in the tree.

If π^* is not discarded while being compared with π, then if $\pi^* \cap R_1(\pi) \neq \emptyset$, $\pi^* \cap R_1(\pi)$ will need to be inserted at $\pi.l$. Similarly, if $\pi^* \cap R_4(\pi) \neq \emptyset$, $\pi^* \cap R_4(\pi)$ will need to be inserted at $\pi.r$. For this reason, the Insert function is recursive. Notice that it may be the case that $\pi.l = \emptyset$ or $\pi.r = \emptyset$. A node π^* is added to the tree if and only if it is inserted at an empty node. Thus, the typical use of the Insert function is to insert a new node π^* at the root node, π_0. Then π^* is either discarded or $\pi^* \cap R_1(\pi)$ and $\pi^* \cap R_4(\pi)$ are inserted at $\pi.l$ and $\pi.r$, respectively. This process repeats recursively until either (i) π^* has been fully discarded, or (ii) all nondominated portions of

π^* have been added to the tree as new nodes. Throughout the remainder of this chapter we will use the notation $\textsc{Replace}(\pi', \tilde{\pi})$ to denote the process of replacing the point or segment associated with $\pi' \in \Pi$ with the point or segment associated with $\tilde{\pi} \in \Pi$ and leaving the tree structure otherwise unchanged. We use the notation $\pi' \leftarrow \tilde{\pi}$ to denote the process of replacing π' and its entire subtree with $\tilde{\pi}$ and its entire subtree. Algorithm 2.1 describes the Insert procedure.

Algorithm 2.1 Inserting a new point or segment, π^*, into the data structure at node π

```
function Insert($\pi^*, \pi$)
    if $\pi^* = \emptyset$ then Return
    if $\pi = \pi_0$ & $\pi_0 \neq \emptyset$ then Rebalance($\pi$)          ▷ $\pi_0$ represents the root node
    if $\pi = \emptyset$ then Replace($\pi, \pi^*$), $\pi$.size $\leftarrow 1$, Update($\pi$)
    else Replace($\pi, \pi \setminus cl\,(R_2(\pi^*))$)
        if $\pi = \emptyset$ then
            if $\pi$.ideal_left $\cap R_2(\pi^*) \neq \emptyset$ then $\pi.l \leftarrow \emptyset$
            if $\pi$.ideal_right $\cap R_2(\pi^*) \neq \emptyset$ then $\pi.r \leftarrow \emptyset$
            RemoveNode($\pi$)
            Insert($\pi^*, \pi$)
        else
            if $\exists\, \pi_1, \pi_2$ s.t. $\pi = \pi_1 \cup \pi_2$ & $cl(\pi_1) \cap cl(\pi_2) = \emptyset$ then
                $\pi_1.l \leftarrow \pi.l$, $\pi_2.r \leftarrow \pi.r$
                $\pi \leftarrow \pi_1$, $\pi.r \leftarrow \pi_2$
                Update($\pi$)
            Insert($\pi^* \cap R_1(\pi), \pi.l$)
            Insert($\pi^* \cap R_4(\pi), \pi.r$)
```

In Algorithm 2.1, the functions RemoveNode and Rebalance refer to the processes of deleting nodes from the tree and rebalancing the tree, respectively. These algorithms will be discussed further in Sections 2.2.2.2 and 2.2.2.3, respectively. The recursive Update function has a node π as input and traverses up the tree from π until reaching the root node. Update then performs two actions: (i) ensures that $\pi.\texttt{size} = (\pi.l).\texttt{size} + (\pi.r).\texttt{size} + 1$ where $(\pi').\texttt{size} = 0$ if and only if $\pi' = \emptyset$, and (ii) ensures that $\pi.\texttt{ideal\_left}$ and $\pi.\texttt{ideal\_right}$ are updated appropriately. After this has been done, if $\pi.p \neq \emptyset$ then $\textsc{Update}(\pi.p)$ is called.

We now introduce a property that is maintained throughout all operations on the tree as described in the remainder of the paper.

Property 2.1. Given an arbitrary node in the tree π, all nodes in the subtree of $\pi.l$ are located completely within $R_1(\pi)$ and all nodes in the subtree of $\pi.r$ are located completely within $R_4(\pi)$.

2.2.2.2 Deletion.

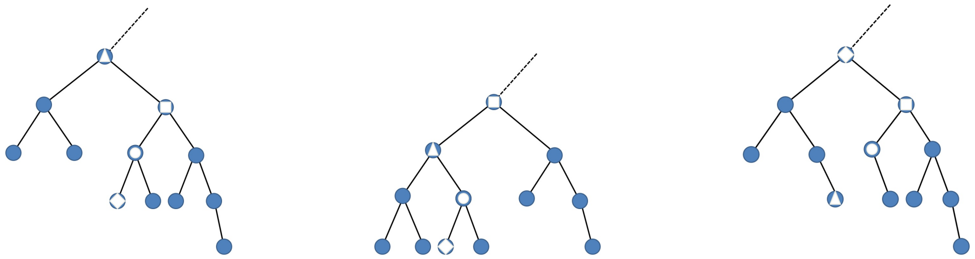
Removing a dominated node from the tree is the next task that frequently needs to be performed. Notice that when a node is deleted, in order for the tree structure to be retained, another node must replace it. This is precisely where the difficulty lies. Usually, when a node is deleted from a quad-tree structure, all nodes contained in the subtree of the deleted node are reinserted in order to maintain proper organization of the tree [91, 92]. Since our quad-tree simplifies to a binary tree, however, we propose something much simpler. Notice that in order for our tree to maintain the appropriate structure, Property 2.1 must be met. For any node π that needs to be removed and replaced, there are precisely two nodes that may replace it and satisfy Property 2.1. They are the right-most node in the subtree of $\pi.l$ and left-most node in the subtree of $\pi.r$. Algorithm 2.2 describes RemoveNode.

Algorithm 2.2 Remove a node that has been shown to be dominated.

```
function RemoveNode($\pi$)
    if $\pi$.size $= 1$ then $\pi \to \emptyset$
    else Define $\tilde{\pi} = \emptyset$
        if $(\pi.l)$.size $> (\pi.r)$.size then $\tilde{\pi} \to$FindRightmostNode($\pi.l$)
        else $\tilde{\pi} \to$FindLeftmostNode($\pi.r$)
        Replace($\pi, \tilde{\pi}$)
        RemoveNode($\tilde{\pi}$)
    if $\pi.p \neq \emptyset$ then Update($\pi.p$)
```

2.2.2.3 Rebalancing.

The final task to perform in maintaining our structure is rebalancing. To maintain balance we use the following strategy of Overmars and Van Leeuwen [67]: for each non-leaf node π, the subtrees of $\pi.l$ and $\pi.r$ must contain no more than $\dfrac{1}{2-\delta}k$ nodes, where k is the number of nodes in π's subtree and δ is a pre-selected value in the open interval $(0, 1)$. Enforcing this requirement causes the depth of the tree to be at most $\log_{2-\delta} t$ where t is the number of nodes in the tree. Now, based on this requirement we develop two rebalancing methods, RebalanceLeft1 and RebalanceLeft2 (and similarly RebalanceRight1 and RebalanceRight2) each of which take a node π as input. In RebalanceLeft2, the left-most node of the subtree of $\pi.r$ is found and is used to replace π. Then π is moved to right-most position of the subtree of $\pi.l$. Notice that RebalanceLeft2 moves a single node from one side of a tree to the other. In certain situations it may be more beneficial to move several nodes from one side of the tree to the other in a single operation. RebalanceLeft1 is designed for this purpose. In RebalanceLeft1, the nodes of the tree are shifted in the following fashion: (i) $\pi.r$ and its right subtree shift up and left to take the place of π and its right subtree, (ii) π and its left subtree shift down and left to become the new left subtree of $\pi.r$, and (iii) the original left subtree of $\pi.r$ is then placed as the new right subtree of π. RebalanceLeft1 and RebalanceLeft2 are illustrated in Figure 2.4.



(a) Original Tree

(b) RebalanceLeft1

(c) RebalanceLeft2

Figure 2.4: Examples of applying rebalancing procedures.

Algorithm 2.3 is used to determine which rebalancing procedure to apply in order to balance the tree. Its correctness is shown in Proposition 5, which is presented in the next section.

Algorithm 2.3 Check to ensure that the balance criterion is met at each node.

```
function Rebalance($\pi$)
    if $(\pi.l)$.size $> 2$ then Rebalance($\pi.l$)
    if $(\pi.r)$.size $> 2$ then Rebalance($\pi.r$)
    if $(\pi.l)$.size $> \frac{\pi.\texttt{size}}{2-\delta}$ then
        if $(\pi.l.l)$.size $\geq \frac{(1-\delta)\pi.\texttt{size}}{2-\delta} - 1$ then RebalanceRight1($\pi$)
        else repeat RebalanceRight2($\pi$) until $(\pi.l)$.size $= \frac{\pi.\texttt{size}}{2-\delta}$
    else if $(\pi.r)$.size $> \frac{\pi.\texttt{size}}{2-\delta}$ then
        if $(\pi.r.r)$.size $\geq \frac{(1-\delta)\pi.\texttt{size}}{2-\delta} - 1$ then RebalanceLeft1($\pi$)
        else repeat RebalanceLeft2($\pi$) until $(\pi.r)$.size $= \frac{\pi.\texttt{size}}{2-\delta}$
```

2.2.3 Performance Guarantees

We now present results about the correctness and complexity of the insertion, deletion, and rebalancing procedures. In this section we will use the notation $\overline{\pi} \in \text{Subtree}(\hat{\pi})$ to denote the case in which $\overline{\pi}$ is a node contained in the subtree of Π which is rooted at $\hat{\pi}$. We assume $\overline{\pi} \in \text{Subtree}(\overline{\pi})$.

Proposition 2.2. Insert removes any portion of a currently stored node π which is dominated by an inserted node π^*.

Proof. Assume $\pi^* \leqq_p \pi \in \Pi$. Since the case in which $\pi = \pi_0$ is trivial, we assume $\pi \neq \pi_0$. Assume WLOG that $\pi \in \text{Subtree}(\pi_0.l)$ and consider the insertion of π^* at π_0. Notice that if $\pi^* \leqq \pi_0$ then in line 5 of Algorithm 2.1, π is replaced by $\pi \setminus cl(R_2(\pi^*)) = \emptyset$. Thus, one of the following occurs: (i) $\pi^* \leqq \pi_0.\texttt{ideal\_left}$ (i.e., $\pi.\texttt{ideal\_left} \cap R_2(\pi^*) \neq \emptyset$) and all nodes contained in $\text{Subtree}(\pi_0.l)$ are removed from Π, or (ii) $\pi^* \nleqq \pi_0.\texttt{ideal\_left}$ and so RemoveNode replaces π_0 with a node $\tilde{\pi} \in \text{Subtree}(\pi_0)$ and the insert procedure recurses. Let $\pi' = \pi^* \cap R_3(\pi)$ which is the portion of π^* that dominates π. Notice that if $\pi^* \nleqq \pi_0$ then $\pi^* \cap cl(R_1(\pi_0))$, which contains π', is inserted at $\pi_0.l$.

The above arguments can be repeated to show that for every $\hat{\pi} \in \Pi$ such that $\pi \in \text{Subtree}(\hat{\pi})$, either: (i) all nodes in $\text{Subtree}(\hat{\pi})$ are removed from Π, or (ii) a portion of π^* containing π' is inserted at $\hat{\pi}$. Note that if a node π'' containing π' is inserted at π, π will be reduced to $\pi \setminus cl(R_2(\pi'')) = \emptyset$. □

Proposition 2.3. Insert adds a portion of an inserted node π^* to the tree if and only if it is not dominated by any node currently stored in the tree.

Proof. Notice that the reverse direction is trivial because if $\tilde{\pi}$ is a portion of π^* not dominated by any $\pi \in \Pi$, then $\tilde{\pi}$ will be inserted at one of the children of every node it is compared against. Thus, since there are a finite number of nodes in the tree, $\tilde{\pi}$ must eventually be inserted at an empty node and added to the tree.

The forward direction is by contraposition. Suppose there is $\pi \in \Pi$ such that $\pi \leqq_p \pi^*$. Let $\pi' = \pi^* \cap cl\,(R_2(\pi))$, (i.e., the portion of π^* that is dominated by π). Assume WLOG that $\pi \in \text{Subtree}(\pi_0.l)$ and consider the insertion of π^* at π_0. Notice that by Property 2.1, $\pi' \subset R_1(\pi_0) \cup R_2(\pi_0)$. If $\pi' \subset R_2(\pi_0)$ then π' will not be added to the tree since only $\pi^* \cap R_1(\pi)$ and $\pi^* \cap R_3(\pi)$ are inserted to the children of π_0. On the other hand, if $\pi' \not\subset R_2(\pi_0)$ then $\pi' \cap R_1(\pi_0) \subset \pi^* \cap R_1(\pi_0)$ and the latter is inserted at $\pi_0.l$.

The above arguments can be repeated to show that for every $\hat{\pi} \in \Pi$ such that $\pi \in \text{Subtree}(\hat{\pi})$, either: (i) π' is thrown out when inserted at a parent or grandparent of $\hat{\pi}$, or (ii) a portion of π^* containing a subset of π' is inserted at $\hat{\pi}$. Notice that if a node π'' containing a subset of π' is inserted at π, neither $\pi'' \cap R_1(\pi)$ nor $\pi'' \cap R_3(\pi)$ will contain any portion of π'. Thus, no portion of π' can be added to the tree. □

Proposition 2.4. Use of the RemoveNode procedure does not violate Property 1.

Proof. Suppose $\pi \in \Pi$ is dominated and must be removed from Π. The case in which π is a leaf node is trivial, so assume that π has at least one child. By Property 1, if $\pi.l \neq \emptyset$ then $\pi \subset R_4(\pi.l)$

and if $\pi.r \neq \emptyset$ then $\pi \subset R_1(\pi.r)$. Thus, if $\pi.l \neq \emptyset$ and π' is the right-most node in Subtree($\pi.l$), then π' is the unique node in Subtree($\pi.l$) such that $\hat{\pi} \subset R_1(\pi')$ for all $\hat{\pi} \in$ Subtree($\pi.l$) $\setminus \pi'$. Similarly, if $\pi.r \neq \emptyset$ and π'' is the left-most node in Subtree($\pi.r$), then π'' is the unique node in Subtree($\pi.r$) such that $\hat{\pi} \subset R_4(\pi'')$ for all $\hat{\pi} \in$ Subtree($\pi.r$) $\setminus \pi''$. Hence, replacing π with either the right-most node in Subtree($\pi.l$) or the left-most node in Subtree($\pi.r$) satisfies Property 1. □

Proposition 2.5. Use of the Rebalance procedure does not violate Property 1.

Proof. We must show that neither RebalanceLeft1 nor RebalanceLeft2 violates Property 1. First consider RebalanceLeft1(π). Note that after this procedure is carried out, $\pi.r$ becomes the root node of the subtree that was once rooted at π. All nodes that were in the subtree of $\pi.r.l$ remain in their original positions relative to $\pi.r$. Now notice that π becomes the left child of $\pi.r$, which does not violate Property 1 since π is completely within $R_1(\pi.r)$. Finally, the entire subtree of $\pi.r.l$ becomes the right subtree of π. Since π is now the left child of $\pi.r$, all of these nodes are still located in the left subtree of $\pi.r$. Furthermore, since these nodes were originally located in π's right subtree, Property 1 is still satisfied.

Now consider RebalanceLeft2(π). In this procedure π is replaced by the left-most node in the subtree of $\pi.r$. We proved that this would not violate Property 1 in the proof of Proposition 4. After this, π is placed as the right child of the node that was previously the right-most node in the subtree of $\pi.l$. We can see that this placement also does not violate Property 1 since all nodes originally within the subtree of $\pi.l$ are completely within $R_1(\pi)$. □

Proposition 2.6. One call of Rebalance(π) satisfies the balance criterion at π.

Proof. WLOG assume that $(\pi.r).\mathtt{size} > \frac{\pi.\mathtt{size}}{2-\delta}$. Now, if $(\pi.r.r).\mathtt{size} < \frac{(1-\delta)\pi.\mathtt{size}}{2-\delta} - 1$ then the proposition is trivially satisfied since in this case RebalanceLeft2 is repeated until $(\pi.r).\mathtt{size} = \frac{\pi.\mathtt{size}}{2-\delta}$. Thus, we focus on the case in which $(\pi.r.r).\mathtt{size} \geq \frac{(1-\delta)\pi.\mathtt{size}}{2-\delta} - 1$. Notice that by the construction of the Rebalance procedure, the subtrees of $\pi.l$ and $\pi.r$ are balanced before that of π. Thus $(\pi.r.r).\mathtt{size} \leq \frac{\pi.\mathtt{size}}{2-\delta}$ because otherwise $(\pi.r.r).\mathtt{size} > \frac{\pi.\mathtt{size}}{2-\delta} > \frac{(\pi.r).\mathtt{size}}{2-\delta}$ which contradicts the fact that the subtree of $\pi.r$ is balanced. Now, suppose that after calling RebalanceLeft1(π), π' is the new root node of the subtree orignally rooted at π. Then the subtree of $\pi'.r$ will be the original subtree of $\pi.r.r$. Thus, since $(\pi.r.r).\mathtt{size} \leq \frac{\pi.\mathtt{size}}{2-\delta}$, the balance criterion will be satisfied for

$\pi'.r$. Also notice that REBALANCELEFT1(π) is only called if $(\pi.r.r).\texttt{size} \geq \frac{(1-\delta)\pi.\texttt{size}}{2-\delta} - 1$. This implies that

$$(\pi.r.r).\texttt{size} \geq \frac{(1-\delta)\pi.\texttt{size} + \pi.\texttt{size} - \pi.\texttt{size}}{2-\delta} - 1$$

$$\Rightarrow \frac{\pi.\texttt{size}}{2-\delta} \geq \pi.\texttt{size} - (\pi.r.r).\texttt{size} - 1.$$

After the procedure is completed, it will be the case that $(\pi'.l).\texttt{size} = \pi.\texttt{size} - (\pi.r.r).\texttt{size} - 1$ where $\pi.\texttt{size}$ is the size of the original subtree rooted at π. Thus, the balance criterion will be satisfied for $\pi'.l$. □

Proposition 2.7. If t is the number of nodes stored in the tree, the worst case complexities of INSERT, REMOVENODE and REBALANCE are $O(t)$, $O(\log t)$ and $O(t^2 \log t)$, respectively.

Proof. INSERT: It is clear that for any comparison between an inserted node π^* and some $\pi \in \Pi$, it is possible that $\pi^* \cap R_1(\pi) \neq \emptyset$ and $\pi^* \cap R_3(\pi) \neq \emptyset$. Thus it is possible for a portion of π^* to be compared with every node in a subtree, implying $O(t)$ complexity.

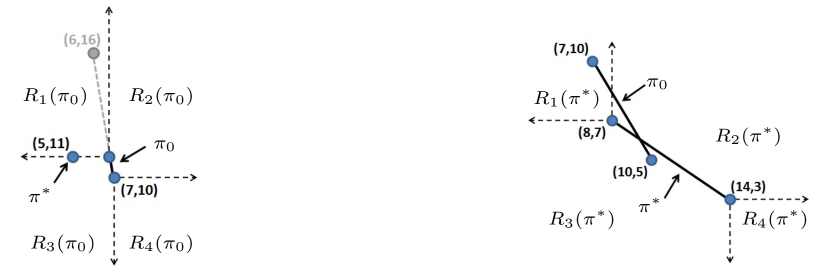
REMOVENODE: Here we assume that the tree is balanced prior to calling REMOVENODE. Recall that when a node π is removed it is replaced with either the left-most node in the subtree of $\pi.r$ or the right-most node in the subtree of $\pi.l$. Since the tree is balanced, finding such a node is clearly an $O(\log t)$ process. If π' is the node replacing π and π' is not a leaf node, then its original position must then be filled using the same process. Note though that in finding the replacement for π' a path through the tree is traversed which begins precisely where the path traversed in finding the replacement for π ended. Thus, even though multiple nodes may need replaced in order for π to be removed, the overall process must result in the traversal of only one path through the tree, resulting in an $O(\log t)$ procedure.

REBALANCE: This requires checking the balance criterion at every node of the tree. Ensuring that the criterion is met at one of these nodes could require repeating the strategy REBALANCELEFT2 up to $\frac{t}{2}$ times. Thus, since REBALANCELEFT2 calls FINDLEFTMOSTNODE, FINDRIGHTMOSTNODE, and UPDATE, which are $O(\log t)$ procedures, the complexity of rebalancing is $O(t^2 \log t)$. □

2.3 Illustrative example

Recall the points and segments specified in Figure 2.2a. We use these points and segments as input to our data structure and show a few of the nontrivial steps of developing our tree. Assume that the solutions shown in the figure are obtained from five separate slice problems and that the Pareto sets of these slice problems, listed in respective order, are: (i) the singleton (1,19), (ii) the piecewise linear curve connecting (1,17) and (9,13), (iii) the piecewise linear curve connecting (6,16) and (11,4), (iv) the singleton (5,11), and (v) the piecewise linear curve connecting (8,7) and (17,2). The points and segments which define these Pareto sets will be inserted into our structure in the order of (iii), (iv), (ii), (v), (i). Piecewise linear curves will be inserted as individual line segments from left to right.

The reader is encouraged to review the pseudocode given previously (particularly Algorithm 2.1). To begin we let $\pi^* \leftarrow (6,16)$ to $(7,10)$ and call $\textsc{Insert}(\pi^*, \pi_0)$. Since $\pi_0 = \emptyset$ we replace π_0 with π^*. Clearly the current tree structure is now a single node. Next we let $\pi^* \leftarrow (7,10)$ to $(10,5)$ and call $\textsc{Insert}(\pi^*, \pi_0)$. Notice that $\pi^* \subset R_4(\pi_0)$ and should be inserted at $\pi_0.r$. Since $\pi_0.r = \emptyset$ this insertion results in π^* being added to the tree. Therefore the tree now contains the root node which has one child to its right. The insertion of the segment connecting $(10,5)$ to $(11,4)$ is analogous.

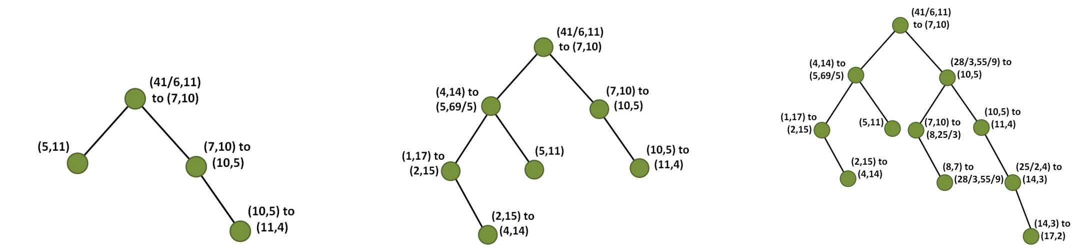


(a) Inserting (5,11).

(b) Inserting $(8,7)$-$(14,3)$.

Figure 2.5: Inserted Segments.

Next consider Pareto set (iv). Let $\pi^* \leftarrow (5,11)$ and call $\textsc{Insert}(\pi^*, \pi_0)$. Observe Figure 2.5a. Clearly we can see that $\pi^* \leqq_p \pi_0$ and thus we remove the dominated portion of π_0 by letting $\pi_0 = \pi_0 \setminus R_2(\pi^*)$. After this has been done, notice that $\pi^* \subset R_1(\pi_0)$. Therefore, since $\pi_0.l = \emptyset$, π^* becomes the left child of π_0. Figure 2.6a shows the tree structure after π^* has been inserted. We leave it to the reader to consider Pareto set (ii). Note, though, that after processing this set the subtree rooted at $\pi_0.l$ needs to be rebalanced. The resulting tree is shown in Figure 2.6b.



(a) Tree after inserting (5,11).

(b) Tree after rebalancing.

(c) Final tree.

Figure 2.6: Tree Structure.

Next we consider the insertion of Pareto set (v). Let $\pi^* \leftarrow (8,7)$ to $(14,3)$ and call Insert(π^*, π_0). Clearly $\pi^* \subset R_4(\pi_0)$ and will therefore be inserted to $\pi_0.r$. Observe from Figure 2.5b that $\pi^* \leqq_p \pi_0.r$. This time, though, the portion of $\pi_0.r$ which is dominated is the center section of the segment. This means that $\pi_0.r$ must be split into two nodes π_1 and π_2. Node π_1 takes the place in the tree where $\pi_0.r$ originally was, and the left subtree of $\pi_0.r$ becomes the left subtree of π_1. Node π_2 becomes the right child of π_1 and the right subtree of $\pi_0.r$ becomes the right subtree of π_2. Now, after this process has been completed, observe that $\pi^* \subset R_4(\pi_1)$ and thus π^* will be inserted to π_2 (which is now $\pi_0.r.r$). Notice that $\pi_0.r.r \leqq_p \pi^*$ and that it is the center portion of π^* that is dominated. Thus the calls to Insert$(\pi^* \cap R_1(\pi_0.r.r), \pi_0.r.r.l)$ and Insert$(\pi^* \cap R_4(\pi_0.r.r), \pi_0.r.r.r)$ will each cause a portion of π^* to be inserted at $\pi_0.r.r.l$ and $\pi_0.r.r.r$ respectively. Since $\pi_0.r.r.l = \emptyset$, $\pi^* \cap R_4(\pi_0.r.r)$ will become $\pi.r.r.l$. Since $\pi_0.r.r.r$ is the segment (10,5) to (11,4), it is clear that another portion of π^* will need to be removed, and then the remainder of π^* will become $\pi_0.r.r.r.r$.

We end our example now because the remaining insertions result in scenarios which are analogous to those that we have now observed. Note that if we were to continue, one more rebalance would be required and the final tree structure would be that found in in Figure 2.6c. Note that this tree structure is dependent on the order of insertion.

2.4 Computational Experiments

We implemented our data structure in the C programming language and performed three tests. The first was designed to test the number of solutions our structure can effectively store and how quickly these solutions can be processed. The second and third were designed to test the utility of our data structure when used alongside the BB algorithms of Belotti et al. [10] and Adelgren and Gupte [1], respectively. All tests were run using Clemson University's Palmetto Cluster. Specifically, an HP SL250s server node with a single Intel E5-2665 CPU core with 16GB of RAM running Scientific Linux 6.4 was used.

In all of these experiments we compare the performance of our structure with that of a dynamic list (L). Like our structure, this list also takes points and segments in $\mathbb{R}^2$ as input and stores only the nondominated subset of all input. When a point or segment is inserted, it is compared with every other stored point or segment. Then, during each comparison dominated solutions are discarded. Such lists have been used for storing nondominated solutions in both the pure integer [92] and mixed-integer cases [63, 101].

2.4.1 Implementation of Rebalance

Recall from Proposition 6 that maintaining a balanced tree is the most costly of the three operations needed to create our structure. It is also the one operation that is unnecessary in order to ensure that we store the correct solutions. For this reason we decided to further consider the rebalancing operations in hopes of finding an alternative implementation that is less computationally costly, but still performs well in practice.

Note that Overmars and Van Leeuwen [67] suggest rebalancing by traversing the path travelled by an inserted solution in the reverse order and checking whether or not the balance criterion is satisfied at each of these nodes. This saves one from having to check the balance criterion at every node in the tree since the only places where it could have been altered are at nodes along this path. In our case, though, when a line segment is inserted into our structure, it often does not remain intact, but is separated into many smaller segments, each traversing its own path through the tree before finally being added. For this reason, rearranging the tree after the insertion of a segment into the tree is troublesome. Hence, we propose a few alternative approaches:

A0 - No rebalancing is used.

A1 - Before allowing a point or segment to be inserted at the root node, check the balance criterion at every node in the tree and rebalance where necessary. Using this approach one is able to guarantee that the balance of the tree is maintained, but its complexity is clearly very high. (Notice that this approach is the implementation used as presented in Algorithms 1 – 5.)

A2 - Periodically check the balance criterion at every node in the tree and rebalance where necessary. For example, we could determine to check the entire tree for balance every time 100 new solutions are added to the tree. This approach significantly decreases the complexity of rebalancing, but eliminates the balance guarantee.

A3 - Another approach could be to check the balance criterion at any node that is currently being inserted at. This approach has a much lower complexity, and would cause balance to be maintained at the root node, and along any frequently travelled paths in the tree. However, again the guarantee of balance is lost.

A4 - A final approach is to combine A2 and A3. Employing A3 alongside A2 may allow one to obtain a well balanced tree by applying A2 far less frequently than when using A2 alone. Clearly the complexity of this approach is higher than that of A3, but it is likely significantly less than that of A2.

We implemented each of these approaches in our first experiment, described in Section 4.2. We utilize approach A2 when performing our other experiments, which are described in Sections 4.3 and 4.4, because for most of our tests A2 performed comparably to A0 in terms of CPU time, but always maintained a more balanced tree.

2.4.2 Experiment 1 – Random Data

2.4.2.1 Setup.

This test has two main purposes:

1. We compare the efficiency of our data structure with that of a dynamic list (which updates via pairwise comparison) when storing nondominated solutions.
2. We determine the number of solutions our structure can take as input and process in a reasonable amount of time.

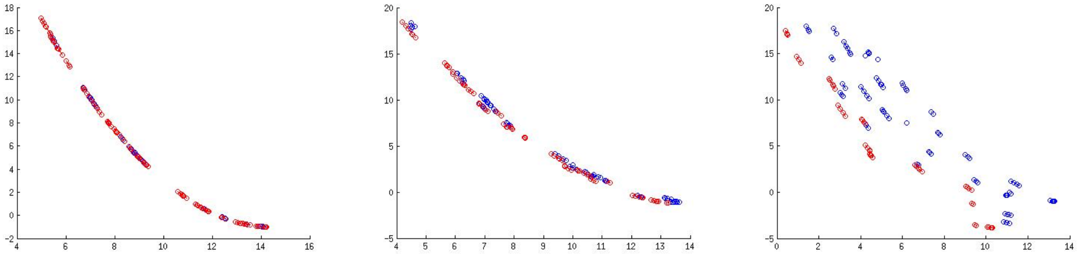
The test consists of repeating the following procedure until N insertions have been made into our structure or the dynamic list. First, generate a random integer $i \in [1, 6]$ and a random number $r_1 \in (0, 10)$. Then, if $i > 1$, for each $j \in \{2, \ldots, i\}$ a random number $c_j \in (0, 1)$ is generated and we define $r_j = r_1 + \sum_{\ell=2}^{j} c_\ell$. Next, for each $j \in \{1, \ldots, i\}$ the following are computed: (i) $y_j = \dfrac{(10.5 - r_j)^2}{5} - k$, and (ii) $x_j = r_j + (5 - k)$. Here k is a dynamic value which is defined as 1 at the start of the test and increases by $\dfrac{\mu}{N}$ each time the above process is repeated. Here

$\mu \in \mathbb{R}$ is a parameter that allows us to determine how much the solutions should "improve" over the course of the test. If $i = 1$, the singleton (x_1, y_1) is inserted into the structure, otherwise the points $(x_1, y_1), ..., (x_i, y_i)$ are arranged in order of increasing x values and then the line segments connecting each adjacent pair of points are inserted into the structure. We performed this test 100 times for each combination of the values $N = 10^4, 10^5, 10^6$ and 10^7 and $\mu = 0, 0.001, 0.01, 0.1, 1$ and 10. We used various values for δ and found that the results were quite similar, but determined to use a value of $\delta = 0.3$. For each test we recorded the time it took to insert all solutions into our structure, the time it took to insert all solutions into the dynamic list, the final depth of our tree, the final number of nodes stored in our tree, and the final number of nodes stored in the dynamic list.

We now explain the significance of μ. Many procedures for determining or approximating the Pareto set of a BOMILP are iterative procedures which attempt to use solutions generated during early iterations to generate better solutions (i.e., solutions which are closer to being Pareto optimal) in later iterations. Such procedures include BB and most heuristic algorithms. Selecting values for μ which are close to zero is intended to replicate generating solutions during one of these procedures in which there is little or no separation between early generated solutions and later generated ones, and thus both early and later generated solutions are likely to be Pareto. Alternatively, selecting large values of μ is intended to replicate generating solutions during one of these procedures in which there is significant separation between early generated solutions and later generated ones, and in which solutions generated later are much more likely to be Pareto than those generated early. We expect to find that our structure performs better for large values of μ since there should be more domination of solutions, therefore requiring less storage. As a visual aid, we include Figure 2.7 which shows an example of solutions generated during this experiment for $\mu = 0.1, 1$ and 10 with $N = 100$. The red solutions are those that are stored by our structure at the end of the test.

2.4.2.2 Implementation Details.

First, recall that as presented, the implementation of our structure performs a check in order to determine whether or not an entire subtree is dominated. If a subtree is found to be dominated, the entire subtree is removed. We found that in practice, however, this implementation does not outperform the implementation in which no check for dominated subtrees is performed, rather dominated nodes are removed one at a time. We feel that there are two drawbacks to the



(a) $\mu = 0.1$

(b) $\mu = 1$

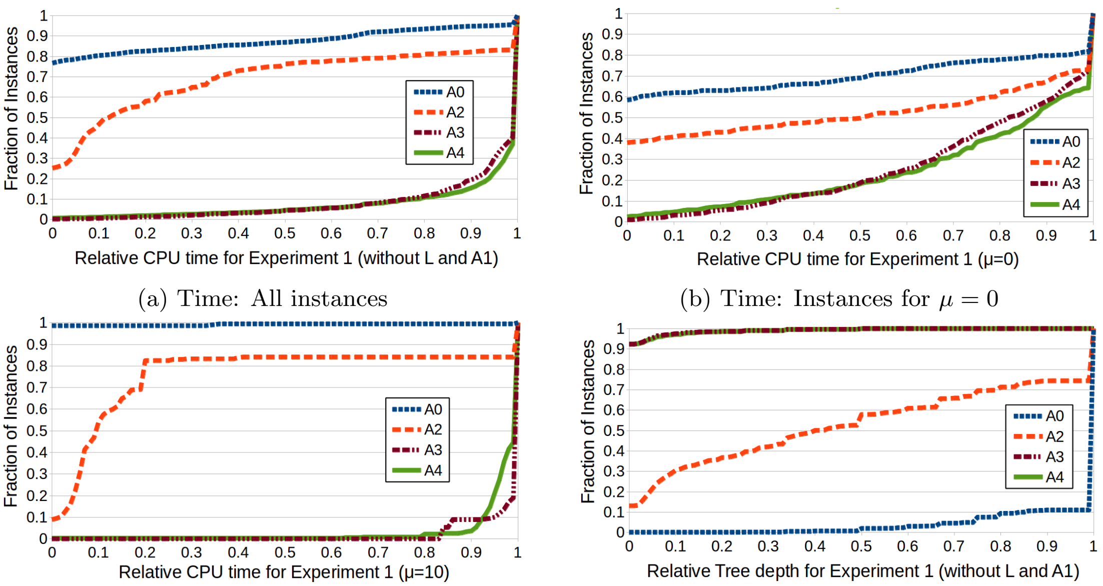
(c) $\mu = 10$

Figure 2.7: Example of solutions generated in Experiment 1 with $N = 100$.

former implementation which are most likely the reasons for this: (i) more information (i.e., an ideal point for each subtree) is stored in each node, and (ii) when new solutions are added to the tree, the UPDATE function must ensure that these ideal points are updated appropriately, which can be a costly procedure. Also notice that the worst case complexity of REMOVENODE remains the same for both implementations. For this reason, we used the latter implementation when performing our tests.

2.4.2.3 Numerical Results.

We present the results obtained from our randomized tests when implementing the five rebalancing approaches discussed in section 4.1.2. When using approach A2, we chose to perform an initial rebalance after 100 new solutions had been added to the structure and then again each time there was a 101% increase in the number of stored solutions. When using approach A4 we again performed an initial rebalance after 100 new solutions had been added to the structure, but this time we did not rebalance again until the number of solutions increased by 800%. The minimum, maximum, and average elapsed times and final depths of the tree resulting from running experiment 1 can be found in Table 2.6 in Section 2.A, though the data for rebalance approaches A0 and A2 are also given in Table 2.1. The performance of the various rebalancing approaches, in terms of CPU time as well as total depth of the tree, is summarized in the performance profiles depicted in Figure 2.8. Note, however, that because the list implementation and rebalancing approach A1 performed so poorly in terms of CPU time when compared to the other approaches, the data associated with both the the list and A1 are omitted from the performance profiles. The average number of nodes stored while running these tests are given in Table 2.2. All averages are reported as geometric



(a) Time: All instances

(b) Time: Instances for $\mu = 0$

(c) Time: Instances for $\mu = 10$

(d) Depth: All instances

Figure 2.8: Performance Profiles for CPU time and tree depth for Experiment 1.

means in Tables 2.1, 2.2 and 2.6. The symbols T and L indicate runs in which our tree structure and the dynamic list were used for storing solutions, respectively. Also, entries in Tables 2.2 and 2.6 which contain dashes are those for which no results are available due to the fact that individual runs took over 12 hours to complete and were therefore terminated. The symbol ⊛, on the other hand, indicates results for which, due to the large amount of time taken for each individual run, we were unable to perform the test 100 times. For these results, each test was instead run 5 times.

Table 2.1: Time and depth of the tree for Experiment 1 (Approaches A0 and A2)

		Rebal	Time (s)			Depth					Rebal	Time (s)			Depth		
μ	N	Type	Min	Avg	Max	Min	Avg	Max	μ	N	Type	Min	Avg	Max	Min	Avg	Max
0	10^4	A0	0.22	0.25	0.33	36	43.8	55	0.001	10^4	A0	0.06	0.06	0.08	24	31.1	42
		A2	0.28	0.35	0.68	16	17.7	20			A2	0.08	0.09	0.12	12	13.8	17
	10^5	A0	11.8	13.1	16.3	36	43.8	55		10^5	A0	1.41	1.65	2.13	26	33.7	44
		A2	13	18.7	63.5	20	21.4	24			A2	1.56	1.77	2.34	14	17	21
	10^6	A0	374	385	393	57	61	65		10^6	A0	36.5	38.6	50.9	35	35.2	54
		A2	351	403	680	22	24.8	27			A2	34.9	36.8	46.4	16	21.5	25
	10^7	A0	7316	7781	9920	66	93	96		10^7	A0	908	1016	1500	33	43.3	55
		A2	5804	6644	8444	24	28.3	44			A2	866	935	1500	21	27	33
0.01	10^4	A0	0.03	0.03	0.04	14	19.9	37	0.1	10^4	A0	0.02	0.02	0.02	11	12.7	17
		A2	0.04	0.05	0.05	10	11.9	14			A2	0.01	0.01	0.03	8	10.3	13
	10^5	A0	0.66	0.69	0.77	18	22.4	34		10^5	A0	0.36	0.36	0.38	13	15.3	18
		A2	0.74	0.75	0.88	12	15.8	20			A2	0.41	0.41	0.43	11	14.3	18
	10^6	A0	14	15.7	23.6	21	25.3	37		10^6	A0	6.43	6.81	8.55	16	18.1	23
		A2	14.4	15.5	19.5	15	20.1	24			A2	6.75	7.03	8.2	14	17.8	21
	10^7	A0	334	352	403	24	28.1	37		10^7	A0	139	140	144	20	20.9	22
		A2	332	342	382	20	24.1	30			A2	140	141	144	18	21.2	26
1	10^4	A0	0.02	0.02	0.02	9	11.1	15	10	10^4	A0	0.01	0.01	0.02	8	9.9	13
		A2	0.02	0.02	0.02	8	9.2	13			A2	0.01	0.02	0.02	7	8.7	13
	10^5	A0	0.24	0.24	0.25	11	12.7	16		10^5	A0	0.21	0.22	0.25	9	10.8	14
		A2	0.26	0.26	0.27	8	12.2	15			A2	0.22	0.23	0.26	8	10.5	13
	10^6	A0	3.63	3.96	4.32	13	15.3	20		10^6	A0	2.46	2.58	2.91	11	12.6	15
		A2	3.77	4.09	4.44	12	15	18			A2	2.52	2.66	2.94	10	12.5	16
	10^7	A0	64.2	65.1	77.7	17	17.9	20		10^7	A0	36.2	36.6	42.8	13	15.1	18
		A2	65.7	66.8	78.2	16	18.2	22			A2	37.2	37.6	43.1	13	15.1	19

All averages are reported as geometric averages.

Table 2.2: Average number of nodes stored in the data structure in Experiment 1.

		Avg # of Nodes				Avg # of Nodes				Avg # of Nodes	
μ	N	T	L	μ	N	T	L	μ	N	T	L
0	10^4	25,737	25,170	0.01	10^4	767	766	1	10^4	188	188
	10^5	198,544	–		10^5	2,369	2,365		10^5	283	283
	10^6	864,145	–		10^6	7,210	7,171		10^6	761	761
	10^7	2,154,322	–		10^7	22,397	–		10^7	2,371	2,368
0.001	10^4	2,368	2,364	0.1	10^4	285	285	10	10^4	133	133
	10^5	7,239	7,200		10^5	766	765		10^5	189	189
	10^6	22,358	⊛ 21,849		10^6	2,366	2,363		10^6	284	284
	10^7	74,165	–		10^7	7,211	⊛ 7,166		10^7	764	764

Each row is an arithmetic average over all rebalancing approaches.

There are several things to notice from Figure 2.8 and Tables 2.1, 2.2 and 2.6. First, notice that in all cases our data structure is able to process inserted solutions much more quickly than the dynamic list. Next, notice that for fixed values of N and μ, A0 typically performs the best in terms of running time, followed by A2 and then A3 and A4. We point out that although Figure 2.8 seems to indicate that A0 significantly outperforms A2 in most cases, one can observe from Table 2.1 that this is not the case. Although A0 does perform better in most cases, there are only a small number of occasions in which the CPU times of A0 and A2 differ by more than a couple of seconds. Furthermore, when these CPU times do differ significantly, A2 often performs better. In terms of maintaining a tree of minimum depth, on the other hand, A3 and A4 typically perform the best. Also notice that for each fixed value of N, the time taken to process inserted solutions decreases as the value of μ increases. Additionally, the larger the value of μ, the closer the time needed for the dynamic list to process the input solutions becomes to the time needed for our tree to process the solutions. By comparing Tables 2.1, 2.2 and 2.6 it is easy to see the correlation between the time taken to process solutions and the number of nodes stored, for both our structure and the dynamic list.

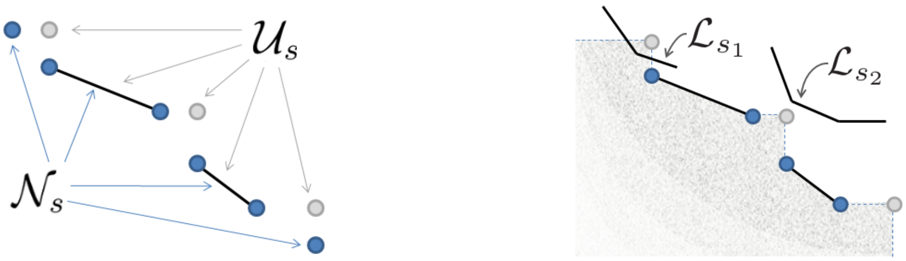
From these results we can see that our data structure can handle the insertion of large sets of solutions, thus we suspect that it can do so without posing a significant overhead on a solution procedure such as BB or a heuristic method.

2.4.3 Experiment 2 – Fathoming in BB of Belotti et al. [10]

In this experiment we solved a variety of BOMILP instances using the BB algorithm of Belotti et al. [10]. We first provide a background on BB procedures for biobjective problems.

The bound sets used for fathoming are not singletons in $\mathbb{R}$ as they are in the single objective case. Instead they are subsets of $\mathbb{R}^2$ formed by taking unions of finitely many continuous convex piecewise linear functions [28]. During each iteration s of BB, a node η_s of the BB tree is considered. The set of Pareto solutions to the LP relaxation of (2.1) associated with η_s provides a lower bound set $\mathcal{L}_s$. The upper bound set $\mathcal{U}_G$, on the other hand, is globally valid to all nodes of the BB tree, although it is generally not known in its entirety until completion of the BB. Therefore, at each iteration s of BB, since $\mathcal{U}_G$ is unknown and cannot be used for fathoming, another set $\mathcal{U}_s = \vartheta(\mathcal{N}_s)$ is used instead, where $\mathcal{N}_s \subset \Omega$ is a set containing no dominated points at iteration s. Now, in order to describe the mapping $\vartheta(\cdot)$, we introduce several definitions. A point $(\kappa_1, \kappa_2) := \kappa \in S \subset \mathbb{R}^2$ is said to be *isolated* in S if $\exists\, \epsilon > 0$ for which $B_\epsilon(\kappa) := \{\hat{\kappa} \in S : \|\kappa - \hat{\kappa}\|_2 < \epsilon\}$ is empty. Given distinct $\overline{\kappa}, \kappa' \in S$ such that $\overline{\kappa}_1 < \kappa'_1$ and $\overline{\kappa}_2 > \kappa'_2$, the point $\kappa^n = (\kappa'_1, \overline{\kappa}_2)$ is called the *local nadir point* with respect to $\overline{\kappa}$ and κ' (note that the above inequalities are strict so that $\kappa^n \neq \kappa'$ and $\kappa^n \neq \overline{\kappa}$). Given a line segment containing points in S, the segment itself is referred to as a *local nadir set*. We now describe the mapping $\vartheta(\cdot)$, which can be used to construct $\mathcal{U}_s$ given $\mathcal{N}_s$. Notice that if a line segment contains no dominated points, then it must have a negative slope. We use the notation $[\kappa^{nw}, \kappa^{se}]$ to denote any such segment, where κ^{nw} and κ^{se} are the segment's north-west and south-east endpoints, respectively. Now, since at any iteration s of BB $\mathcal{N}_s \subset \Omega$ contains no dominated points, each of its elements must be either an isolated point or a line segment with a negative slope. For each point $\kappa \in \mathcal{N}_s$ consider κ_1, and for each segment $[\kappa^{nw}, \kappa^{se}] \in \mathcal{N}$ consider κ_1^{nw}. Arrange the elements of $\mathcal{N}_s$ in increasing order of these values. Then for each pair of adjacent elements $(\varepsilon_1, \varepsilon_2) \in \mathcal{N}_s$, if the south-east-most point of ε_1 is not equal to the north-west-most point of ε_2, calculate the local nadir point with respect to these two points and add it to a set $\mathcal{N}'_s$. Note that if ε_i for $i \in \{1, 2\}$ is a point and not a segment, then its north-west-most and south-west-most points are simply ε_i itself. Now let $\mathcal{N}''_s$ be the set of local nadir sets in $\mathcal{N}_s$. Then $\vartheta(\mathcal{N}_s) := \mathcal{N}'_s \cup \mathcal{N}''_s$ and thus $\mathcal{U}_s = \mathcal{N}'_s \cup \mathcal{N}''_s$. Figure 2.9a illustrates the relationship between $\mathcal{N}_s$ and $\mathcal{U}_s$.

One of the fathoming rules presented by Belotti et al. [10] states that at iteration s of BB a node η_s can be fathomed if $\mathcal{L}_s$ is separable from $\mathcal{U}_s$, i.e., $\mathcal{L}_s \cap (\mathcal{U}_s - \mathbb{R}^2_+) = \emptyset$. This is essentially the extension of the well known "fathoming by bound dominance" rule for single-objective problems to the biobjective case. Figure 2.9b shows examples of lower bound sets $\mathcal{L}_{s_1}$ and $\mathcal{L}_{s_2}$. Notice that the locations of these sets show that η_{s_1} cannot be fathomed but η_{s_2} can. Clearly, efficient fathoming



(a) Examples of $\mathcal{N}_s$ and $\mathcal{U}_s$.

(b) η_1 cannot be fathomed, but η_2 can.

Figure 2.9: Examples of bound sets and fathoming rules.

depends on the choice of $\mathcal{N}_s$ used to construct $\mathcal{U}_s$ since good approximations of $\mathcal{U}_G$ at each iteration of BB can aid in fathoming a large number of nodes.

At iteration s of BB, let $\mathcal{F}_s$ be the set of all $\psi \in \Omega$ discovered during iterations $1, \ldots, s-1$ of BB. Then at iteration s, the best choice for $\mathcal{N}_s$ is the nondominated subset of $\mathcal{F}_s$. Finding this set can be cumbersome though. Until now, there seem to have been only two approaches used:

Dynamic List: Each time $\psi \in \Omega$ is found, store it in a list and then remove dominated points and segments by performing a pairwise comparison between all stored solutions. After completion of the pairwise comparison the set of stored solutions is precisely $\mathcal{N}_s$ and $\mathcal{U}_s$ can be constructed as $\vartheta(\mathcal{N}_s)$. Such lists have been used by Mavrotas and Diakoulaki [63], Vincent et al. [101].

Predetermined subset of Ω_P: Before beginning BB, a preprocessing phase is used to generate a set $\mathcal{N} \subset \Omega_P$ and this $\mathcal{N}$ is used as the set $\mathcal{N}_s$ at every iteration s of BB. Therefore a single set $\mathcal{U} = \vartheta(\mathcal{N})$ is used for fathoming throughout the entire BB. Note that one way to generate $\mathcal{N}$ is to use the ϵ-constraint method [cf. 27], i.e., solve the MILP

$$\min_{x,y} \left\{ c_1^\top x + d_1^\top y \colon \; c_2^\top x + d_2^\top y \leq \epsilon, (x,y) \in P_I \right\}$$

for various values of ϵ. For each value of ϵ such that this MILP is feasible, its solution (x_ϵ, y_ϵ) belongs to X_E and thus corresponds to a point in Ω_P. Then $\mathcal{N}$ is the union of all Pareto points found this way. Although this option eliminates the need for updating via pairwise comparison, its effectiveness is highly dependent on the number of initial points generated and the ability to solve single objective MILPs very fast. Therefore, there is a clear tradeoff between computational time versus quality of the upper bound set. This method was used by Belotti et al. [10].

2.4.3.1 Setup.

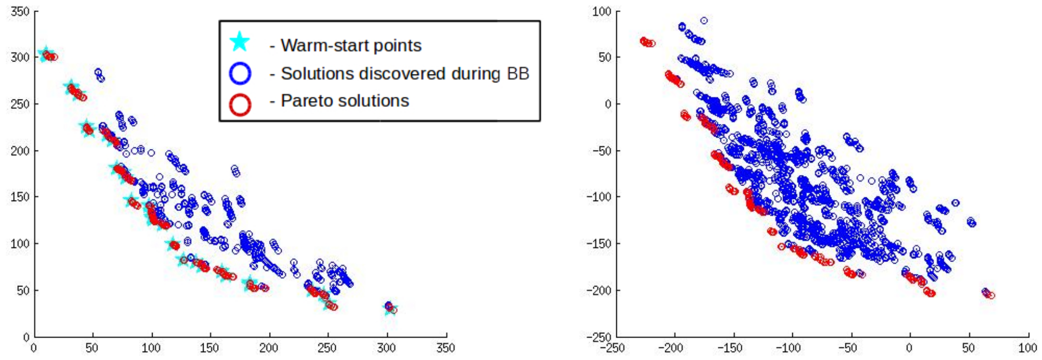
We experimented on the instances from Belotti et al. [10] and Boland et al. [15] and present results on all instances that took between 10 seconds and 8 hours to solve. The Boland et al. [15] instances are divided into two types of problems, but during preliminary tests the BB code ran into numerical issues with the second type and so we did not experiment with them. Each instance was solved three times – once using our structure in order to generate the upper bound set at each iteration of the BB, once using a dynamic list in order to generate these sets, and once using a predetermined subset of Ω_P to generate a single upper bound set which was used for fathoming throughout the BB.

2.4.3.2 Implementation Details.

First we point out that when utilizing the predetermined subset of Ω_P, the ϵ-constraint method was used to generate $M \leq M^*$ points from Ω_P before beginning the BB, where M^* is a user-selected upper bound on the number of these points that are generated. Notice, though, that these M points can still be useful in the cases when either our structure or a dynamic list is being used alongside the BB. By inserting these points into either structure at the start of the BB, the procedure can be "warm-started," increasing the frequency and efficiency of fathoming.

Initially we solved several instances using our structure both with and without warm-starting. However, the results we obtained without warm-starting were very poor, and are therefore not reported. Notice that warm-starting allows solutions which are "far" from the set of Pareto-optimal solutions to be discarded early in the BB, and therefore fewer nodes of the BB tree are explored. As a visual aid, observe Figure 2.10 which shows solutions generated during the BB procedure when warm-starting is and is not used.

Notice that warm-starting provides M points which are in most cases well dispersed throughout Ω_P. Therefore we attempted an implementation in which these points M were inserted into our data structure in such a way that BB begins with a perfectly balanced tree. We felt that this may allow us to turn off the rebalancing procedures and in turn solve each instance more quickly. However, the results obtained from this implementation did not provide any increased efficiency and so we report results from the original implementation that uses rebalancing A2.



(a) Using warm-start with $M^* = 100$

(b) Not using warm-start

Figure 2.10: An instance from [10] with 80 variables and 80 constraints.

We solved each instance using various values for M^*, ranging from 10 on small instances to 3000 on large ones. "Good" choices for the value of M^* seem to be highly dependent on the size and difficulty of the instance being solved.

2.4.3.3 Numerical Results.

Tables 2.3 and 2.4 present the results of this experiment. The problem sizes in these tables are reported as the number of variables, which in all cases also equalled the number of constraints. We use P, T, and L to represent the implementations of the predetermined set of Ω_P, our tree structure, and the dynamic list, respectively. When we refer to a "tree" in these results, we mean our data structure, not the BB tree. There are 30 instances for each of the two problem sizes in Belotti et al. [10] and we report geometric averages for them in Table 2.3. For the type I instances in Boland et al. [15], there are 5 instances available for each problem size. Individual results for each solved instance are provided in Table 2.4 with geometric averages over the 5 instances displayed in bold. We were unable to solve the size 320 instances using this BB within 8 hours.

While conducting this experiment we found that the CPU time utilized by each data structure was significantly less than the total CPU time used during BB. For this reason we decided to employ the profiler *gprof* [38] to measure the percentage of total BB time which was spent executing functions associated with each data structure. These percentages are also presented in Tables 2.3 and 2.4. Note, however, that when averaging these percentages we used the arithmetic average since some values were measured as zero.

Table 2.3: Results of Experiment 2 for instances from Belotti et al. [10].

		Time (s)			% Time In Structure		# of BB Nodes Fathomed			# of Inserts		Final # of Nodes Stored		Final Depth
Size	M	P	T	L	T	L	P	T	L	T	L	T	L	of Tree
60	10	25.6	17.1	16.8	0.04	0.07	516	342	342	2,191	2,190	75.0	76.3	12.8
	25	15.3	13.4	13.2	0.04	0.06	309	271	271	1,421	1,421	75.8	78.4	13.4
	50	14.6	13.5	13.4	0.03	0.09	273	254	254	1,246	1,246	76.0	79.6	13.4
80	10	65.8	43.9	43.2	0.01	0.03	705	492	492	2,743	2,743	89.0	90.1	9.6
	25	44.1	38.0	37.6	0.01	0.03	488	428	428	2,102	2,102	89.5	91.6	10.1
	50	41.2	38.5	38.0	0.01	0.03	430	403	403	1,869	1,869	89.6	93.3	9.7

Each row is an average over 30 instances. Geometric averages are used for all columns except % time in structure.

As it is difficult to know ahead of time what value of M^* is most appropriate for solving a given BOMILP, we ran each of these instances for a large number of values for M^*, but for the sake of space only report results for 3 of these values – small, medium and large.

Table 2.4: Results of Experiment 2 for instances from Boland et al. [15].

		Time (s)			% Time In Structure		# of BB Nodes Fathomed ($\times 10^2$)			# of Inserts ($\times 10^3$)		Final # of Nodes Stored		Final Depth
Size	M	P	T	L	T	L	P	T	L	T	L	T	L	of Tree
80	50	163	98	100	0.16	1.86	53	18	18	67	69	1,062	1,061	19
		130	86	86	0.15	1.31	27	13	13	120	117	673	694	12
		169	113	116	0.14	2.28	54	21	21	64	64	933	954	11
		134	97	99	0.22	2.43	40	13	13	58	58	928	947	11
		86	76	76	0.06	0.63	36	20	20	47	54	746	759	13
		133	**93**	**94**	**0.14**	**1.70**	**41**	**17**	**17**	**68**	**70**	**856**	**872**	**12**
	200	41	40	40	0.08	1.12	13	10	10	37	37	1,052	1,104	15
		33	33	33	0.18	0.84	9	7	7	38	38	676	720	17
		50	51	52	0.13	1.39	16	12	12	27	26	916	992	12
		44	42	43	0.13	2.02	13	8	8	36	36	930	990	15
		42	41	41	0.04	0.44	17	13	13	30	30	757	799	14
		41	**41**	**42**	**0.11**	**1.16**	**13**	**10**	**10**	**34**	**33**	**855**	**909**	**14**
	300	34	34	36	0.05	1.44	11	9	9	34	35	1,058	1,132	16
		31	32	32	0.15	0.87	8	7	7	42	42	676	718	12
		45	44	45	0.00	1.44	14	11	11	24	24	927	1,031	13
		41	41	41	0.08	2.32	11	8	8	35	35	938	1,012	13
		36	35	36	0.05	0.56	14	12	12	28	28	758	823	11
		37	**37**	**38**	**0.06**	**1.32**	**11**	**9**	**9**	**32**	**32**	**860**	**930**	**12**
160	500	8,541	9,663	9,651	0.04	0.32	743	616	616	3,274	3,277	2,794	2,922	52
		12,533	14,731	14,651	0.02	0.14	1,541	1,193	1,193	2,806	2,816	2,976	3,079	39
		8,188	8,593	8,522	0.03	0.20	1,053	935	936	1,857	1,864	2,725	2,859	17
		4,759	5,357	5,327	0.04	0.47	437	363	363	1,319	1,344	6,156	6,310	26
		1,723	1,787	1,779	0.03	0.22	196	168	168	572	575	3,043	3,130	23
		5,906	**6,512**	**6,480**	**0.03**	**0.27**	**635**	**530**	**531**	**1,667**	**1,678**	**3,354**	**3,476**	**29**
	2,000	5,640	6,214	6,164	0.03	0.74	533	509	510	2,484	2,486	2,802	3,280	28
		8,146	9,031	9,013	0.02	0.59	1,008	947	947	2,156	2,165	2,994	3,354	30
		6,413	6,613	6,569	0.03	0.32	888	861	864	1,449	1,475	2,765	3,214	33
		3,238	3,444	3,471	0.04	1.65	341	320	322	958	1,006	6,159	6,783	18
		1,326	1,355	1,360	0.03	0.70	149	141	141	461	462	3,083	3,410	13
		4,173	**4,444**	**4,439**	**0.03**	**0.80**	**475**	**451**	**452**	**1,279**	**1,298**	**3378**	**3824**	**23**
	3,000	5,527	5,989	6,013	0.04	1.14	516	500	500	2,417	2,418	2,796	3,510	24
		7,650	8,496	8,540	0.03	0.96	967	921	922	2,090	2,118	2,982	3,519	24
		6,191	6,424	6,386	0.03	0.48	870	849	851	1,396	1,423	2,757	3,432	24
		3,167	3,345	3,414	0.05	2.39	332	315	316	916	941	6,179	7,110	22
		1,298	1,325	1,330	0.04	1.05	144	138	138	449	451	3,089	3,560	28
		4,040	**4,288**	**4,311**	**0.03**	**1.20**	**461**	**443**	**443**	**1,238**	**1,254**	**3,376**	**4,038**	**24**

Each row in the table is for a single instance. Times reported in column P do not include the extra time that is required to find the Pareto set upon termination of a BB using a predetermined subset of Ω_P.

As can be seen in Tables 2.3 and 2.4, for low values of M^* computation time is significantly reduced when using our structure or the dynamic list in place of the predetermined subset of Ψ_P. As M^* increases, though, the computation time resulting from using the predetermined subset of Ψ_P approaches the time resulting from using our tree or the list. In some cases, when M^* is high enough or instances are sufficiently large and challenging, the computation time resulting from using the predetermined subset of Ψ_P is actually less than the time resulting from using our tree or the list. There are two main reasons for this. First, larger values of M^* result in initial bound sets which more closely approximate the Pareto set of a given instance. Thus, the additions made to this set throughout BB by our tree or the list do not aid in fathoming as much as they do for low values of M^*. The second reason why the computation time when using the predetermined subset of Ψ_P is occasionally less than that of our tree or the list is that the difficulty of fathoming in the BB of Belotti et al. [10] increases significantly as the number of stored integer solutions increases. Hence, for larger instances more time is spent processing each BB node. We point out, however, that the Pareto set is not readily available upon termination when BB is implemented using the predetermined subset of Ω_P. Instead, the set of all integer feasible solutions is stored and a post-processing phase is needed in order to determine the Pareto set. The implementations using our tree and the list data structure, on the other hand, do have the Pareto set readily available upon termination. The times reported in column P in Table 2.4 do not include the extra time that is required to compute the Pareto set upon termination of BB and this additional time can be quite significant. Thus the marginal advantage in CPU time that a predetermined subset may yield is effectively neutralized when comparing the total time required to solve a BOMILP instance and compute its entire Pareto set. Finally, we also note that the number of nodes fathomed from the BB tree is generally lower when using our structure or the dynamic list as opposed to the predetermined subset of Ω_P. This is because nodes of the BB are being fathomed earlier, or higher, in the tree, therefore causing fewer nodes to be explored.

When comparing the results of this experiment to those of our first experiment one may wonder why our tree structure does not significantly outperform the list in all cases. On inspecting the number of solutions inserted to our structure versus the final number stored, we found that a value of $\mu \approx 100$ (see Section 2.4.2.1 and Figure 2.7) could be associated with most of the solved instances. This value of μ indicates that there is a high level of separation amongst the solutions generated during BB and therefore a large fraction of generated solutions ends up being dominated and hence not stored. Thus for BB experiments, there is not a significant difference between our structure and the list in terms of the time needed to process the data, at least for the sizes of instances considered in this experiment. Support for this can be found by observing the percentages of CPU time spent in each data structure as reported in Tables 2.3 and 2.4.

2.4.4 Experiment 3 – Fathoming in BB of Adelgren and Gupte [1]

To further test the utility of our proposed data structure alongside BB, we performed another set of tests in which BOMILP instances of BOMILP were solved using the BB algorithm of Adelgren and Gupte [1]. We chose this BB algorithm mainly because it utilizes a dynamic data structure, such as a list or our tree, not only for storing found solutions, but also to check for domination of bound sets and hence, for fathoming a node in the BB tree. Thus this BB algorithm exploits the data structure proposed in this chapter to the fullest possible extent. In contrast, the BB of Belotti et al. [10] employs a different set of fathoming rules. A secondary motivation for testing with this new BB is that, as reported in Adelgren and Gupte [1], it is able to solve larger instances in under 8 hours, including the type 2 instances from Boland et al. [15].

2.4.4.1 Setup.

Each instance was solved twice, once using our structure in order to generate the upper bound set at each iteration of BB, and once using a dynamic list in order to generate these sets. As with Experiment 2, the BOMILP instances were taken from Belotti et al. [10] and Boland et al. [15] and both problem types are included for the latter. The BB of Adelgren and Gupte [1] uses warm starting by default. However, the user does not need to specify a value for M^*. Instead, an iterative procedure is used to generate solutions in Ω_P. This procedure is terminated when the number of new solutions in Ω_P generated in a given iteration falls below a specified threshold.

2.4.4.2 Numerical Results.

Table 2.5 presents the results of this experiment. The first two rows report averages as was done in Table 2.3 whereas individual results are given for the Boland et al. [15] instances with averages reported in bold. Note that all averages in Table 2.5 are reported as geometric means with the exception of arithmetic means being used for the % time spent in data structure since some of these values were very close to zero.

There are a few pieces of key information that are important to notice in Table 2.5. First, recognize that for almost every instance displayed (and especially for large challenging instances) the percentage of total BB time which was spent maintaining our tree structure is significantly lower than the percentage of total BB time spent maintaining the list. Note that this is the same pattern

Table 2.5: Results of Experiment 3.

Instances		Time (s)		% Time In Structure		BB Nodes Explored		# of Insertions		Final # of Nodes Stored		Final Depth
From	Size	T	L	T	L	T	L	T	L	T	L	of Tree
† (averaged)	60	5	5	0.01	0.08	46	45	203	204	82	76	6
	80	12	12	0.01	0.03	54	53	242	245	103	95	6
‡ (Type 1)	80	10	10	0.33	3.41	187	187	5,168	4,674	1,321	1,160	13
		9	8	0.44	1.67	145	145	3,751	4,150	820	745	10
		18	17	0.12	4.91	423	424	7,686	7,776	1,181	1,048	12
		9	9	0.36	4.50	173	171	4,536	4,822	1,252	1,041	11
		5	5	0.38	1.69	173	171	2,997	2,935	907	826	11
		9	**9**	**0.32**	**3.23**	**194**	**195**	**4,584**	**4,633**	**1,077**	**951**	**11**
	160	116	123	0.16	4.13	745	745	18,024	17,779	3,698	3,208	13
		174	195	0.11	4.85	720	748	27,340	26,470	4,445	3,915	14
		87	92	0.12	4.21	476	486	17,354	17,611	3,044	2,645	13
		290	349	0.12	10.21	1,504	1,514	52,384	54,194	7,777	6,696	14
		74	78	0.13	3.63	430	431	12,679	13,456	3,252	2,754	13
		131	**143**	**0.12**	**5.40**	**697**	**707**	**22,438**	**22,712**	**4,173**	**3,609**	**13**
	320	3,588	4,251	0.05	6.93	3,214	3,464	188,292	198,860	18,128	15,724	16
		6,994	9,361	0.04	11.46	5,263	5,405	284,914	286,246	27,848	24,026	17
		6,512	8,217	0.04	7.37	5,154	5,703	251,172	219,755	18,531	9,560	16
		7,012	8,152	0.04	6.09	5,774	5,981	274,737	281,625	21,643	18,873	27
		3,505	4,400	0.06	6.53	3,678	3,874	180,288	187,390	17,968	15,662	25
		5,257	**6,514**	**0.04**	**7.67**	**4,503**	**4,771**	**231,675**	**231,168**	**20,519**	**16,057**	**19**
‡ (Type 2)	800	2	2	0.27	0.00	41	41	314	340	87	79	7
		2	3	0.00	0.00	38	40	355	382	89	95	7
		6	5	0.00	0.20	79	79	597	537	136	115	7
		8	8	0.13	0.37	122	121	715	756	167	174	8
		4	**4**	**0.10**	**0.14**	**62**	**62**	**467**	**479**	**115**	**110**	**7**
	1250	10	10	0.21	0.41	126	125	786	824	201	195	8
		26	29	0.04	0.11	301	305	1,893	1,956	237	264	8
		26	26	0.12	0.24	252	259	1,569	1,731	319	319	9
		39	37	0.03	0.44	290	281	1,887	1,698	399	360	10
		23	**23**	**0.10**	**0.30**	**229**	**229**	**1,448**	**1,475**	**279**	**277**	**8**
	2500	179	186	0.04	0.31	615	607	2,006	2,086	452	474	10
		364	415	0.02	0.22	1,004	1,036	2,754	2,800	548	536	10
		593	664	0.02	0.26	1,279	1,361	3,482	3,457	634	636	12
		1,335	1,270	0.01	0.25	2,315	2,377	4,516	4,372	652	678	10
		477	**505**	**0.02**	**0.26**	**1,162**	**1,194**	**3,052**	**3,065**	**565**	**575**	**10**

† – Belotti et al. [10], ‡ – Boland et al. [15].

we observed in experiment 2, and it clearly indicates that as instance size and difficulty increases, the impact each data structure has on total CPU time will also increase. This is also supported by the data in Table 2.5. Observe the significant difference in total BB time for the largest instances, particularly Type 1 instances of size 320 from Boland et al. [15]. For these instances we see that the tree provides a decrease in total running time of between 10 and 25 percent. Note that there is one Type 2 instance of size 2500 from Boland et al. [15] in which the list outperformed our tree. However, even for this group of instances our tree performed better on average. Also note that the % of BB time used by our tree decreases as the problem size increases whereas the opposite is true for the list. This indicates that the absolute amount of time used by our tree is far more consistent than that used by the list. In all, the results of experiment 3 clearly indicate that using our tree alongside BB is advantageous. Furthermore, the results also indicate that as BB techniques for BOMILP improve and larger instances are able to be solved, the advantage of using our tree will continue to grow.

2.5 Conclusion

In this work we have introduced a new data structure, in the form of a modified binary tree, that is able to efficiently store sets of nondominated solutions of BOMILPs. Until now similar structures have only been used in the pure integer case. We provide an extension for the more difficult mixed-integer case. We showed this structure performs with a worst case guarantee of $O(t^2 \log t)$ where t is the number of stored nodes. We tested the practical value of our data structure with three experiments. The results show that our structure provides a more efficient method for storing solutions to BOMILP than other prevalent techniques. They also show that our structure is a very useful tool when used alongside branch-and-bound methods for solving BOMILPs.

Generalizing the ideas proposed in this chapter to the multiobjective mixed integer case is one line of future research, although we remark that this extension is not immediate because of the difficulty of doing comparisons and bound dominance in dimensions greater than two. We also recognize that within the scope of biobjective programming, there may be ways to increase the efficiency of our data structure, as explained next. Recall that each node of our structure may store either a point or a line segment. It is possible that in certain cases our structure stores several segments that all belong to a single piecewise linear curve. Therefore it may be beneficial to extend the functionality of our structure so that entire piecewise linear curves can be stored in a single node. Notice that in some cases this may allow for a significant reduction of the size of the tree and thus allow the structure to be populated and maintained more quickly. The reason that we did not implement our structure in this fashion is that for the BOMILP solution techniques we are familiar with, segments are generated one at a time and in general connecting segments are not generated sequentially. Also, for the specific instances we solved, it was not often that a significant number of connected line segments generated from the same slice problem were Pareto optimal. Implementing such refinements to the proposed data structure remains an avenue of future work.

2.A Appendix A: Detailed results for Experiment 1

Table 2.6: Time and depth of the tree for Experiment 1.

		Rebal	Time (s)			Depth		
μ	N	Type	Min	Avg	Max	Min	Avg	Max
0	10^4	A0	0.22	0.25	0.33	36	43.8	55
		A1	1.77	2.14	3.54	16	16.4	17
		A2	0.28	0.35	0.68	16	17.7	20
		A3	0.27	0.34	0.60	16	17.1	18
		A4	0.27	0.34	0.61	16	17.1	18
		L	252	309	438	–	–	–
	10^5	A0	11.8	13.1	16.3	36	43.8	55
		A1	489	623	1,010	19	20.1	21
		A2	13.0	18.7	63.5	20	21.4	24
		A3	13.1	19.6	70.8	20	20.7	22
		A4	13.2	19.6	79.7	20	20.7	22
		L	–	–	–	–	–	–
	10^6	A0	374	385	393	57	61.0	65
		A1	–	–	–	–	–	–
		A2	351	403	680	22	24.8	27
		A3	381	432	698	22	23.0	23
		A4	385	435	678	22	23.0	24
		L	–	–	–	–	–	–
	10^7	A0	7,316	7,781	9,920	66	93.0	96
		A1	–	–	–	–	–	–
		A2	5,804	6,644	8,444	24	28.3	44
		A3	6,312	7,322	9,326	24	24.6	26
		A4	6,400	7,424	38,709	24	24.6	25
		L	–	–	–	–	–	–
0.001	10^4	A0	0.06	0.06	0.08	24	31.1	42
		A1	0.33	0.34	0.40	12	12.2	13
		A2	0.08	0.09	0.12	12	13.8	17
		A3	0.08	0.09	0.14	12	13.1	14
		A4	0.08	0.09	0.14	12	13.1	14
		L	2.72	2.83	2.94	–	–	–
	10^5	A0	1.41	1.65	2.13	26	33.7	44
		A1	9.8	11.6	17.9	13	14.0	15
		A2	1.56	1.77	2.34	14	17.0	21
		A3	1.69	1.97	2.87	14	14.9	16
		A4	1.68	1.97	2.89	14	15.0	16
		L	205	309	824	–	–	–
	10^6	A0	36.5	38.6	50.9	35	35.2	54
		A1	291	343	641	15	15.9	17
		A2	34.9	36.8	46.4	16	21.5	25
		A3	38.3	40.5	51.6	16	16.8	18
		A4	38.3	40.2	52.0	16	16.8	18
		L	–	–	–	–	–	–
	10^7	A0	908	1,016	1,5000	33	43.3	55
		A1⊛	24,166	25,245	27,624	17	17.7	18
		A2	866	935	1,500	21	27.0	33
		A3	963	1,031	1,646	18	18.7	20
		A4	967	1,038	1,595	18	18.7	20
		L	–	–	–	–	–	–

Table 2.7: (Continuation of Table 2.6.)

		Rebal	Time (s)			Depth		
μ	N	Type	Min	Avg	Max	Min	Avg	Max
0.01	10^4	A0	0.03	0.03	0.04	14	19.9	37
		A1	0.13	0.13	0.14	10	10.3	12
		A2	0.04	0.05	0.05	10	11.9	14
		A3	0.05	0.05	0.05	11	11.2	12
		A4	0.04	0.04	0.05	11	11.2	12
		L	0.49	0.50	0.53	–	–	–
	10^5	A0	0.66	0.69	0.77	18	22.4	34
		A1	3.54	3.60	3.86	12	12.1	14
		A2	0.74	0.75	0.88	12	15.8	20
		A3	0.85	0.88	0.93	12	13.1	14
		A4	0.85	0.88	0.93	13	13.1	14
		L	30.1	30.7	33.1	–	–	–
	10^6	A0	14.0	15.7	23.6	21	25.3	37
		A1	99.6	117	217	13	14.0	15
		A2	14.4	15.5	19.5	15	20.1	24
		A3	16.8	18.2	24.9	14	14.9	16
		A4	16.8	18.3	24.8	14	14.9	16
		L	2,079	2,926	6,602	–	–	–
	10^7	A0	334	352	403	24	28.1	37
		A1	2,935	3,109	3,607	15	15.5	17
		A2	332	342	382	20	24.1	30
		A3	373	384	438	16	16.5	17
		A4	377	386	451	16	16.5	18
		L	–	–	–	–	–	–
0.1	10^4	A0	0.02	0.02	0.02	11	12.7	17
		A1	0.06	0.06	0.06	8	8.7	10
		A2	0.01	0.01	0.03	8	10.3	13
		A3	0.02	0.02	0.02	9	9.4	10
		A4	0.01	0.01	0.02	9	9.4	10
		L	0.13	0.14	0.15	–	–	–
	10^5	A0	0.36	0.36	0.38	13	15.3	18
		A1	1.38	1.39	1.45	10	10.2	11
		A2	0.41	0.41	0.43	11	14.3	18
		A3	0.51	0.52	0.53	10	11.2	12
		A4	0.51	0.51	0.53	10	11.2	12
		L	5.22	5.28	5.37	–	–	–
	10^6	A0	6.43	6.81	8.55	16	18.1	23
		A1	35.7	38.5	54.1	12	12.1	14
		A2	6.75	7.03	8.20	14	17.8	21
		A3	8.59	8.98	10.7	12	13.1	14
		A4	8.56	8.68	10.7	12	13.1	14
		L	306	360	856	–	–	–
	10^7	A0	139	140	144	20	20.9	22
		A1	997	1,012	1,086	13	14.0	15
		A2	140	141	144	18	21.2	26
		A3	168	169	178	14	14.9	16
		A4	168	170	176	14	15.0	16
		L⊛	20,903	24,326	29,097	–	–	–

Table 2.8: (Second continuation of Table 2.6.)

		Rebal	Time (s)			Depth		
μ	N	Type	Min	Avg	Max	Min	Avg	Max
1	10^4	A0	0.02	0.02	0.02	9	11.1	15
		A1	0.04	0.04	0.04	7	8.0	9
		A2	0.02	0.02	0.02	8	9.2	13
		A3	0.02	0.02	0.02	8	8.6	10
		A4	0.02	0.02	0.02	8	8.6	10
		L	0.09	0.09	0.10	–	–	–
	10^5	A0	0.24	0.24	0.25	11	12.7	16
		A1	0.63	0.63	0.65	8	8.7	10
		A2	0.26	0.26	0.27	8	12.2	15
		A3	0.33	0.33	0.34	9	9.4	10
		A4	0.32	0.32	0.34	9	9.4	10
		L	1.42	1.44	1.57	–	–	–
	10^6	A0	3.63	3.96	4.32	13	15.3	20
		A1	13.8	15.7	17.7	10	10.3	12
		A2	3.77	4.09	4.44	12	15.0	18
		A3	5.21	5.70	6.23	10	11.2	12
		A4	5.18	5.72	6.31	10	11.2	12
		L	52.7	65.9	81.7	–	–	–
	10^7	A0	64.2	65.1	77.7	17	17.9	20
		A1	357	365	494	12	12.1	14
		A2	65.7	66.8	78.2	16	18.2	22
		A3	86.1	87.7	104	13	13.1	14
		A4	85.7	87.3	106	13	13.1	14
		L	1,481	3,229	6,211	–	–	–
10	10^4	A0	0.01	0.01	0.02	8	9.9	13
		A1	0.03	0.03	0.03	7	7.3	8
		A2	0.01	0.02	0.02	7	8.7	13
		A3	0.02	0.02	0.02	7	8.0	9
		A4	0.02	0.02	0.02	7	8.0	9
		L	0.06	0.06	0.07	–	–	–
	10^5	A0	0.21	0.22	0.25	9	10.8	14
		A1	0.40	0.45	0.57	8	8.1	9
		A2	0.22	0.23	0.26	8	10.5	13
		A3	0.26	0.27	0.31	8	8.8	10
		A4	0.25	0.28	0.31	8	8.8	10
		L	0.92	1.03	1.17	–	–	–
	10^6	A0	2.46	2.58	2.91	11	12.6	15
		A1	6.39	6.78	8.02	8	8.7	10
		A2	2.52	2.66	2.94	10	12.5	16
		A3	3.37	3.52	3.93	9	9.5	11
		A4	3.31	3.48	3.97	9	9.5	11
		L	14.3	15.9	20.0	–	–	–
	10^7	A0	36.2	36.6	42.8	13	15.1	18
		A1	139	141	177	10	10.2	12
		A2	37.2	37.6	43.1	13	15.1	19
		A3	52.2	52.9	61.7	10	11.1	12
		A4	51.7	52.2	62.3	10	11.1	12
		L	528	531	561	–	–	–

Chapter 3

A Branch-and-Bound Algorithm for BOMILP

[The contents of this chapter include material from a paper entitled "A branch-and-bound method for biobjective mixed integer linear programs," which will be submitted to the journal *Mathematical Programming Computation* in August of 2016; the authors are N. Adelgren and A. Gupte.]

3.1 Introduction

Biobjective mixed integer linear programs (BOMIP) have the following form,

$$\min_{x} \left\{ \begin{array}{l} f_1(x) := (c^1)^\top x \\ f_2(x) := (c^2)^\top x \end{array} \right\} \quad \text{s.t.} \quad x \in X_I := \left\{ x \in \mathbb{R}^m_+ \times \mathbb{Z}^n_+ : Ax \leq b, l_i \leq x_i \leq u_i \right\}.$$

A BOMIP is considered solved when the set of so-called Pareto optimal points has been discovered; see [27] for details. Applications of multiobjective programming can be found in a variety of disciplines, including engineering, business, and management. Many problems require the use of discrete quantities, and thus BOMIP is an important class of problems. Exact algorithms for BOMIP are proposed in [11, 16]; the former is an extension of the standard branch-and-bound (BB) algorithm for MIPs whereas the latter is a search method in the (f_1, f_2)-space that recursively solves MIPs. Additional BB techniques for subclasses of BOMIP are proposed in [71, 90].

In this work we introduce a branch-and-bound framework for general BOMIP. Our main body of work is in developing new node processing techniques that account for the challenges of biobjective problems. We are also first to discuss the extension of a variety of dual presolve techniques to the multiobjective setting and we provide empirical evidence of the utility of these methods. Additionally, we show that one of these presolve techniques, probing on integer variables, can also be used alongside branching in order to develop tighter bounds and reduce the computational effort needed for BB. Through the use of locally valid cutting planes and the solution of single objective MIPs, we are able to propose methods for developing tighter dual bounds than have previously been proposed. We also introduce a new, challenging set of test instances which we develop from classical single objective instances available in the MIPlib 2010 library. As many of these instances are quite challenging, we propose the first technique for measuring a "duality gap" for multiobjective BB which relies on the computation of an approximated version of the well-known Hausdorff distance between two subsets of $\mathbb{R}^2$.

3.2 Preliminaries

3.2.1 Definitions and Notation

The idea of optimality for single objective optimization is replaced with the idea of *efficiency* in multiobjective problems. Given distinct $x', x'' \in X_I$, we say that $y' = \mathbf{f}(x')$ *dominates* $y'' = \mathbf{f}(x'')$ if $\mathbf{f}(x') \leq \mathbf{f}(x'')$. We denote this relationship as $y' \succ y''$. We then say that $x \in X_I$ is *efficient* if there is no $x' \in X_I$ such that $\mathbf{f}(x') \succ \mathbf{f}(x)$. The set of efficient solutions is denoted by X_E. Let $Y_I = \{y = \mathbf{f}(x) \in \mathbb{R}^p : x \in X_I\}$. Then $y \in Y_I$ is called *Pareto optimal* if its preimage is efficient. Y_N denotes the set of Pareto optimal points.

For each $k \in \{1, 2\}$ we also define the set

$$Y_I^k := \{y \in Y_I : y_k = \min_{x \in X_I} \{f_k(x) : f_i(x) \leq f_i(\hat{x}) \text{ for all } i \neq k,\ \hat{x} \in X_I\}\} \tag{3.1}$$

and let y_I^k denote an arbitrary element of Y_I^k. Note that for either X_I, Y_I, or Y_I^k, if we drop the "I" subscript we are indicating the continuous relaxation of the given set. Also, if we add a subscript "s", then it means that the set is associated with node s of the BB tree. We use $\mathcal{OS}$ to denote the *objective space*, the smallest rectangle in $\mathbb{R}^2$ containing Y. Given $S \subseteq \mathcal{OS} \subseteq \mathbb{R}^2$, the *ideal point* of S, denoted S^{ideal}, is the point $y \in \mathbb{R}^2$ for which $y_k = \min_{y \in S}\{y_k\}$ for each $k \in \{1, 2\}$. Additionally, we denote the nondominated subset of S as S^{nd}. Throughout this work we also utilize the set $\mathbb{R}^2_{\geq} = \{y \in \mathbb{R}^2 : y_1 \geq 0, y_2 \geq 0\}$.

We assume background in branch-and-cut algorithms for single objective problems; see [60] for a survey. One of the key differences and challenging aspects of BOMIP versus MIP is the concept of primal and dual bound sets, which we explain next.

3.2.2 Bound sets for BOMIP

In biobjective BB, unlike the single objective case, primal and dual bounds are not scalars but rather subsets of $\mathbb{R}^2$. These bound sets were introduced by [28]. We treat the dual bound set as a single polyhedron in $\mathbb{R}^2$ and the primal bound set as a finite union of polyhedra in $\mathbb{R}^2$. Note that this deviates from the traditional view of bound sets which defines them in terms of the boundary of these polyhedra. However it is straightforward to see that equivalent fathoming rules exist for each definition.

Consider any arbitrary node s of the BB tree. We use $\mathcal{L}_s$ to denote the locally valid dual bound set generated from Pareto solutions of the BOLP relaxation at this node; this dual bound can be calculated as $\mathcal{L}_s = Y_s + \mathbb{R}^2_{\geq}$. Let $\mathcal{N}_s$ denote the current nondominated set of solutions in $\mathcal{OS}$ that correspond to some feasible integer solutions in X_I; this set $\mathcal{N}_s$ is the nondominated subset of $\cup_{s' \in \mathscr{N}_s}(Y_{s'})_I$, where $\mathscr{N}_s$ is the set of nodes processed prior to s. The globally valid primal bound generated from the solutions in $\mathcal{N}_s$ is $\mathcal{U}_s := \mathcal{N}_s + \mathbb{R}^2_{\geq}$. Using these sets, the most basic idea of fathoming for BOMIP is: node s can be fathomed if $\mathcal{L}_s \subseteq \mathcal{U}_s$. Figure 3.1 shows an example of these bound sets. Notice that node s_2 can be fathomed but we cannot say anything about fathoming node s_1 since $\mathcal{L}_{s_1} \nsubseteq \mathcal{U}_s$.

Recognize that, similar to the single objective case, correct fathoming rules are essential for any BB algorithm to solve BOMIP to Pareto optimality. However, as can be imagined from Figure 3.1, fathoming is even more crucial and computationally intensive for BOMIPs since it involves checking inclusion and intersection of polyhedral sets as opposed to comparing scalar values in the MIP case. Thus, the majority of the computational effort in multiobjective BB is spent processing a node s of the BB tree, in particular checking various fathoming rules.

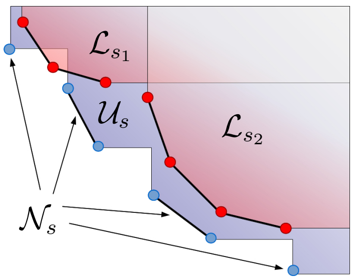


Figure 3.1: Biobjective bound sets

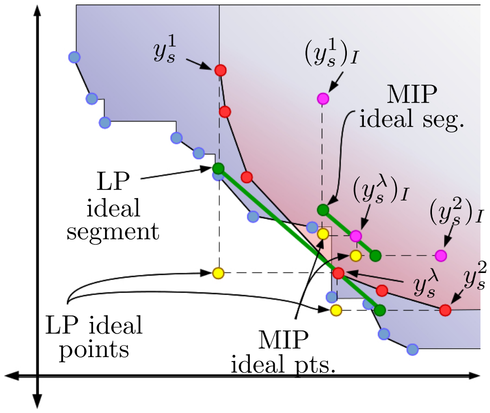


Figure 3.2: Ideal points and segment

3.3 Node processing

Processing a node consists of three basic steps: (i) Generate a valid dual bound; (ii) Check a fathoming rule to determine whether or not s can be eliminated from the search tree; (iii) Optionally, if s is not fathomed in (ii), generate a tighter dual bound and repeat (ii). Figure 3.2 provides a visual example of how one might carry out these three steps. Most of the fathoming rules for biobjective BB are designed to check whether or not $\mathcal{U}_s$ dominates $(Y_s)_I$ by exploiting the transitivity of dominance. First, a set $\mathcal{T}$ is generated such that $\mathcal{T} \succ (Y_s)_I$. Then if $\mathcal{U}_s \succ \mathcal{T}$, $\mathcal{U}_s \succ (Y_s)_I$ and s can be fathomed. Otherwise, a tighter bound on $(Y_s)_I$ is needed. The first bound we use is a set of two ideal points which we obtain by solving three single objective LPs; one for each f_k and an one with a weighted sum objective f_λ in which the weights, denoted λ^s, are given by the normal vector of the line segment H_s passing through y_s^1 and y_s^2. We begin with these points because it is straightforward to determine whether or not $\mathcal{U}_s$ dominates a singleton. In Figure 3.2 these points are labelled "LP ideal points." Notice that they are not dominated. Consider the intersection of $(Y_s)^{ideal} + \mathbb{R}^2_{\geq}$ and the line with normal vector λ^s passing through y_s^λ. Recognize that this intersection, which we denote H_s^λ, is also a valid dual bound. In Figure 3.2 the resulting line segment is labelled "LP ideal segment," but is not dominated. A tighter bound can next be found by explicitly generating $\mathcal{L}_s$. In Figure 3.2 this is the set indicated by the red points, which is again not dominated. After generating $\mathcal{L}_s$, one cannot hope to find a tighter bound on $(Y_s)_I$ resulting from LP solutions. Instead, one can solve single objective MIPs to generate elements of $(Y_s)_I$ and use these elements to form a valid dual bound. We first generate ideal points in the same way as before, but use single objective MIPs rather than LPs.

In Figure 3.2 these points are labelled "MIP ideal points." Yet again they are not dominated. We can then consider the intersection of $((Y_s)_I)^{ideal} + \mathbb{R}^2_{\geq}$ and the line with normal vector λ^s passing through $(y^\lambda_s)_I$, which we denote $\tilde{H}^\lambda_s$. This intersection forms another valid dual bound. In Figure 3.2 the resulting line segment is labelled "MIP ideal segment" and is dominated. Hence, s can be fathomed in this example.

We now formally introduce a proposition outlining the fathoming rules we employ in this work. For use in this proposition, we introduce some additional notation. For each $k \in \{1, 2\}$, define

$$\mathcal{P}^k_s := \left(\cup_{i \neq k} \tilde{y}^i_s\right) \cup \tilde{y}^\lambda_s. \tag{3.2}$$

Then let

$$\mathcal{P}_s := (\mathcal{P}^1_s)^{ideal} \cup (\mathcal{P}^2_s)^{ideal}. \tag{3.3}$$

Additionally, given any $\mathcal{I} \subset \{1, 2, \lambda\}$, define

$$D^\mathcal{I}_s := \cup^2_{k=1} \left(\left(\mathcal{P}^k_s \setminus \{\cup_{i \in \mathcal{I}} y^i_s\}\right) \cup \{\cup_{i \in \mathcal{I} \setminus \{k\}} (y^i_s)_I\}\right)^{ideal}. \tag{3.4}$$

Recognize that $\mathcal{P}_s$ represents the sets of ideal points obtained from LP solutions, while and $D^\mathcal{I}_s$ represents a set of ideal points obtained from a mixture of LP and MIP solutions.

Proposition 3.1 (Fathoming Rules)**.** Node s can be fathomed if:

0. $\mathcal{L}_s \subset (Y_s)_I$

1a. $(\mathcal{U}_s \succ \mathcal{P}_s)$

2a. $\left(\mathcal{U}_s \succ H^\lambda_s\right)$

1b. $(\mathcal{U}_s \succ D^\mathcal{I}_s$ for some $\mathcal{I} \subset \{1, 2, \lambda\})$

2b. $(\mathcal{U}_s \succ \tilde{H}^\lambda_s)$.

3. $\mathcal{L}_s \subseteq \mathcal{U}_s$

Proof.

Rule 0: The result is trivial since $\mathcal{L}_s$ is integer feasible.

Rule 1a: The result is trivial since by construction $\mathcal{P}_s \succ \mathcal{L}_s$ and so $\mathcal{U}_s \succ \mathcal{L}_s$.

Rule 2a: The result is trivial since by construction $\tilde{H}^\lambda_s \succ \mathcal{L}_s$ and so $\mathcal{U}_s \succ \mathcal{L}_s$.

Rule 1b: By construction, for any $\mathcal{I} \subset \{1, 2, \lambda\}$, $D_s^{\mathcal{I}}$ dominates every $(y_s)_I \in (Y_s)_I$ and thus $D_s^{\mathcal{I}}$ is a valid dual bound at node s. The desired result follows.

Rule 2b: By construction H_s^λ dominates every $(y_s)_I \in (Y_s)_I$ and thus H_s^λ is a valid dual bound at node s. The desired result follows.

Rule 3: The result is trivial. □

Proposition 3.1 outlines five fathoming rules. Rule 0 expresses the idea of fathoming due to optimality, while the remainder of the rules indicate situations in which s can be fathomed due to bound dominance.

Before we outline the process we use for processing a node s, we briefly discuss another important task that ought to be carried out while processing node s: Updating $\mathcal{N}_s$. We do this in two ways: (i) add each integer-feasible line segment discovered while checking Fathoming Rule 0 to $\mathcal{N}_s$, and (ii) for each discovered $x^* \in X_I$, generate the nondominated subset of

$$\mathcal{Y}(x^*) := \{y = \mathbf{f}(x) : x \in X, x_i = x_i^* \text{ for all } i \in \{m+1, \ldots, m+n\}\} \tag{3.5}$$

and add each defining line segment of this set to $\mathcal{N}_s$. Consider the latter of these strategies. Observe that the feasible set of $\mathcal{Y}(x^*)$ can be interpreted as a leaf node of the BB tree, which we denote $s(x^*)$. Hence, the $\mathcal{Y}(x^*) + \mathbb{R}^2_{\geq} = \mathcal{L}_{s(x^*)}$. This leads to a need for generating the nondominated subset of $\mathcal{L}_s$, i.e. $\mathcal{L}_s^{nd}$. Typical techniques for generating $\mathcal{L}_s^{nd}$ include the multiobjective simplex method and the parametric simplex algorithm (PSA) [27]. However, the multiobjective simplex method is far more robust than is necessary for biobjective problems. Also, we found in practice that using the PSA often resulted in many basis changes yielding the same extreme point of $\mathcal{L}_s$ in $\mathcal{OS}$. Since much work is done during the PSA to determine the entering and exiting variables, we found that generating $\mathcal{L}_s^{nd}$ using the PSA required a significant amount of computational effort. We decided to use an alternative method for generating $\mathcal{L}_s^{nd}$ which relies on sensitivity analysis. We first solve the single objective LP using objective f_2 to obtain y_s^2. Next we create the LP

$$\mathscr{P}_s(\alpha) := \min\{f_1(x) + \alpha f_2(x) : x \in X_s\} \tag{3.6}$$

and then carry out the procedure outlined in Algorithm 3.1.

In lines 3 and 4 of Algorithm 3.1 we compute the south-east and north-west most extreme points of $\mathcal{L}_s^{nd}$, respectively. The while loop beginning on line 5 is then used to sequentially compute

Algorithm 3.1 Generate $\mathcal{L}_s^{nd}$
Input: Node s.
Output: A set $\mathcal{B}$ containing all defining line segments of $\mathcal{L}_s^{nd}$.

```
1: function GenerateDualBd($s$)
2:     Set $\mathcal{B} = \emptyset$.
3:     Solve the LP $\min\{f_2(x) : x \in X_s\}$ to obtain $y_s^2$.
4:     Solve $\mathscr{P}_s(0)$ to obtain solution $x^*$ and set $y = \mathbf{f}(x^*)$.
5:     while $y \neq y_s^2$ do
6:         Use sensitivity analysis to obtain an interval $[\alpha', \alpha'']$ such that $x^*$ is optimal to $\mathscr{P}_s(\alpha)$ for all $\alpha \in [\alpha', \alpha'']$.
7:         Set $x^* = \text{argmin}\{\mathscr{P}_s(\alpha'' + \epsilon)\}$ for sufficiently small $\epsilon > 0$.
8:         if $\mathbf{f}(x^*) \neq y$ then
9:             Add the line segment connecting $\mathbf{f}(x^*)$ and $y$ to $\mathcal{B}$. Update $y$ to be $\mathbf{f}(x^*)$.
10:    Return $\mathcal{B}$.
```

adjacent extreme points of $\mathcal{L}_s^{nd}$ in a west to east pattern, until the south-east most extreme point is rediscovered. Each line segment joining a pair of adjacent extreme points of $\mathcal{L}_s^{nd}$ is stored and the set of all computed segments is returned at the end of the procedure.

Recognize from Proposition 3.1 that Fathoming Rules 0 and 3 each impose a condition on $\mathcal{L}_s$ and therefore require knowledge of $\mathcal{L}_s^{nd}$ in order to be employed. We note, however, that for each of these rules it is often unnecessary to generate $\mathcal{L}_s^{nd}$ entirely. In particular, the generation of $\mathcal{L}_s^{nd}$ should cease if: (i) one is checking Fathoming Rule 0 and a defining line segment of $\mathcal{L}_s^{nd}$ is generated that is not integer feasible, or (ii) one is checking Fathoming Rule 3 and a defining line segment of $\mathcal{L}_s^{nd}$ is generated that is not contained in $\mathcal{U}_s$. Hence, the procedures in Algorithm 3.1 can be modified in order to develop strategies for checking Fathoming Rules 0 and 3. These strategies are outlined in Algorithms 3.2 and 3.3, respectively.

Algorithm 3.2 follows almost the same procedure as Algorithm 3.1, except it terminates prematurely on line 10 if a line segment is computed that is not integer feasible. Algorithm 3.3 also follows almost the same procedure as Algorithm 3.1. However, this procedure terminates prematurely on line 5 or 12 if a point or line segment is computed that is not dominated by $\mathcal{U}_s$. We have now built the tools necessary to present our proposed procedure for processing a node s. We do so in Algorithm 3.4.

Line 2 of Algorithm 3.4 is an optional procedure in which we can generate locally valid cutting planes to strengthen the representation of X_s if so desired. We then compute y_s^1 and y_s^2 on line 3. We then check to see if either of these solutions are integer feasible, and if they are, we generate the dual bound associated with the integer solution in order to update $\mathcal{N}_s$. Furthermore,

Algorithm 3.2 Fathoming Rule 0
Input: Node s and solutions y_s^1 and y_s^2.
Output: 1 if node s should be fathomed, 0 otherwise.

```
function FR_0($s, y_s^1, y_s^2$)
    $y_s^1$ is the solution to $\mathscr{P}_s(0)$. Let $x^*$ represent the preimage of $y_s^1$. Set $y = y_s^1$.
    if $y = y_s^2$ then return 1
    else
        while $y \neq y_s^2$ do
            Use sensitivity analysis to obtain an interval $[\alpha', \alpha'']$ such that $x^*$ is optimal to $\mathscr{P}_s(\alpha)$ for all $\alpha \in [\alpha', \alpha'']$.
            Update $x^*$ to be the solution of $\mathscr{P}_s(\alpha'' + \epsilon)$ for an arbitrarily small $\epsilon > 0$.
            if $\mathbf{f}(x^*) \neq y$ then
                Let $\mathbb{S}$ represent the line segment connecting $\mathbf{f}(x^*)$ and $y$.
                if $\mathbb{S} \not\subset (Y_s)_I$ then return 0
                else Update $y$ to be $\mathbf{f}(x^*)$.
        return 1
```

Algorithm 3.3 Fathoming Rule 3
Input: Node s and solutions y_s^1 and y_s^2.
Output: 1 if node s should be fathomed, 0 otherwise.

```
function FR_3($s, y_s^1, y_s^2$)
    $y_s^1$ is the solution to $\mathscr{P}_s(0)$. Let $x^*$ represent the preimage of $y_s^1$. Set $y = y_s^1$.
    if $y = y_s^2$ then
        if $\mathcal{U}_s \succ y$ then return 1
        else return 0
    else
        while $y \neq y_s^2$ do
            Use sensitivity analysis to obtain an interval $[\alpha', \alpha'']$ such that $x^*$ is optimal to $\mathscr{P}_s(\alpha)$ for all $\alpha \in [\alpha', \alpha'']$.
            Update $x^*$ to be the solution of $\mathscr{P}_s(\alpha'' + \epsilon)$ for an arbitrarily small $\epsilon > 0$.
            if $\mathbf{f}(x^*) \neq y$ then
                Let $\mathbb{S}$ represent the line segment connecting $\mathbf{f}(x^*)$ and $y$.
                if $\mathcal{U}_s \not\succ \mathbb{S}$ then return 0
                else Update $y$ to be $\mathbf{f}(x^*)$.
        return 1
```

if both solutions are integer feasible, we check Fathoming Rule 0 on line 6. On line 7 we compute the value λ_s, the value of the weights on the objectives so that the level curves of f_λ have the same slope as the line segment joining y_s^1 and y_s^2. We then solve the LP associated with f_λ. If the solution is integer feasible, we again update $\mathcal{N}_s$ as before. On line 9 we check whether or not y_s^1, y_s^2 and y_s^λ are dominated by $\mathcal{U}_s$. If they are, we proceed to check Fathoming Rules 1a, 2a, and 3. Otherwise, we solve the MIP associated with f_λ and f_k for each $k \in \{1, 2\}$ such that the ideal point $(\mathcal{P}_s^k)^{ideal}$ is not dominated by $\mathcal{U}_s$. On lines 21 and 22 we utilize the solutions of each MIP to (optionally) add local cuts to X_s and update $\mathcal{N}_s$. Finally, we check Fathoming Rules 1b and 2b.

Algorithm 3.4 Process node s

```
function ProcessNode($s$)
    Compute valid cutting planes for $(X_s)_I$ and add them to the description of $X_s$.
    for $k \in \{1, 2\}$ do Solve $\min\{f_k(x) : x \in X_s\}$ to find optimal solution $\bar{x}^k$ and generate $y_s^k \in Y_s^k$.
        if $y_s^k \in (Y_s)_I$ then let $N = $ GenerateDualBd($s(\bar{x}^k)$) and set $\mathcal{N}_s = (\mathcal{N}_s \cup N)^{nd}$.
    if $y_s^1, y_s^2 \in (Y_s)_I$ then
        if FR_0($s, y_s^1, y_s^2$) $= 1$ then Fathom $s$, STOP!                    (Fathoming Rule 0)
    Calculate $H_s$ and $\lambda^s$ using $y_s^1$ and $y_s^2$. Solve $\min\{f_\lambda(x) : x \in X_s\}$ to find optimal solution $\bar{x}^\lambda$ and generate $y_s^\lambda \in Y_s^\lambda$.
    if $y_s^\lambda \in (Y_s)_I$ then let $N = $ GenerateDualBd($s(\bar{x}^\lambda)$) and set $\mathcal{N}_s = (\mathcal{N}_s \cup N)^{nd}$.
    if $\mathcal{U}_s \succ y_s^1$, $\mathcal{U}_s \succ y_s^2$ and $\mathcal{U}_s \succ y_s^\lambda$ then
        if $\mathcal{U}_s \succ \mathcal{P}_s$ then Fathom $s$, STOP!                 (Fathoming Rule 1a)
        else
            Calculate $\tilde{H}_s^\lambda$.
            if $\mathcal{U}_s \succ \tilde{H}_s^\lambda$ then Fathom $s$, STOP!          (Fathoming Rule 2a)
            else
                if FR_3($s, y_s^1, y_s^2$) $= 1$ then Fathom $s$, STOP!         (Fathoming Rule 3)
    else
        Define the set $\mathcal{I} = \emptyset$.
        for $k \in \{1, 2\}$ do
            if $\mathcal{U}_s \not\succ (\mathcal{P}_s^k)^{ideal}$ then add $(\{1, 2\} \setminus \{k\}) \cup \{\lambda\}$ to $\mathcal{I}$
        for each $k \in \mathcal{I}$ do solve the MIP $\min\{f_k(x) : x \in (X_s)_I\}$ to find optimal solution $\hat{x}^k$ and obtain $(y_s^k)_I \in (Y_s^k)_I$.
            Add a local cut to $X_s$ which lies on the level curve of $f_k$ associated with the best found dual solution.
            Let $N = $ GenerateDualBd($s(\hat{x}^k)$) and set $\mathcal{N}_s = (\mathcal{N}_s \cup N)^{nd}$.
        if $\mathcal{U}_s \succ D_s^{\mathcal{I}}$ then Fathom $s$, STOP!                 (Fathoming Rule 1b)
        else if $\lambda \in \mathcal{I}$ then
            Calculate $H_s^\lambda$.
            if $\mathcal{U}_s \succ H_s^\lambda$ then Fathom $s$, STOP!                (Fathoming Rule 2b)
```

We now proceed to Section 3.4 in which we discuss the extension of the remaining major aspects of single objective BB to the biobjective setting.

3.4 Biobjective BB

In this section we discuss the specifics of how the different components of single objective BB – presolve/preprocessing, node processing, and branching, can each be extended to the biobjective setting. We then briefly discuss optional additions to our basic biobjective BB procedure.

3.4.1 Presolve/Preprocessing

It has been shown in a variety of works that examining the structure of an instance of single objective MIP prior to solving it, and utilizing information found during this examination to simplify the structure of the instance often has a significant impact on the time and effort needed to solve that instance. It has also been shown that knowledge of feasible solutions for an instance of MIP can have quite an impact on solution time. Hence, it makes sense to extend the techniques used in these procedures to the biobjective case. For the discussion that follows we distinguish the idea of simplifying an instance of BOMIP based on its problem structure from the idea of determining a set of initial integer-feasible solutions. We refer to the first as *presolve* and the latter as *preprocessing.* We propose a procedure which is carried out in three phases: (i) Presolve phase 1, (ii) Preprocessing and (iii) Probing on variables.

3.4.1.1 Presolve Phase 1

When presolve techniques are utilized for single objective MIP, both primal and dual information is used. Fortunately, the primal information of a BOMIP instance is no different than its single objective counterpart and thus primal presolve techniques can be applied directly to it. However, due to the presence of an additional objectives, one must take more care in order to utilize dual information when employing a biobjective presolve strategy.

We extend a few single objective presolve techniques to the multiobjective case (though we implement them for the biobjective case); in particular, we discuss duality fixing [60] and the exploitation of singleton and dominating columns [33]. The ideas are straightforward generalizations of what is already known for MIPs. In the following discussion the element of matrix A in row r and column j is denoted by a_{rj}.

Proposition 3.2 (Duality fixing)**.** Suppose there exists a column j with $c^k_j \geq 0$ and $a_{ij} \geq 0$ for all k, i. If $\ell_j > -\infty$, then $X_E \subseteq \{x \colon x_j = \ell_j\}$. Similarly, if there exists a column j with $c^k_j \leq 0$ and $a_{ij} \leq 0$ for all k, i, then $X_E \subseteq \{x \colon x_j = u_j\}$ assuming $u_j < \infty$.

Proof. It is well known (see Theorem 4.5 of [27]) that x^* is efficient for the given BOMIP if and only if there exists ϵ such that x^* is optimal to the problem:

$$\min_x \{f_1(x)\} \; s.t. \; \{x \in X_I : f_k(x) \leq \epsilon_k \text{ for all } k \neq 1\} \tag{3.7}$$

Hence, every efficient solution to the given BOMIP can be obtained by solving (3.7) for some ϵ. Recognize that if the conditions given in this proposition hold, then single objective duality fixing can be applied to (3.7). This shows that every efficient solution to the given BOMIP can be obtained by solving the modified version of (3.7) in which variable fixing has been performed. □

Proposition 3.3 (Singleton Columns)**.** For every row r in the system $Ax \leq b$, define $\overline{J}(r) := \{j \in \{1, \ldots, m\} : a_{rj} > 0, c_j^k < 0 \ \forall k, a_{ij} = 0 \ \forall i \neq r\}$ and $\overline{U}_r := \sum_{j \in \overline{J}(r)} a_{rj}\ell_j + \sum_{j \notin \overline{J}(r), a_{rj} > 0} a_{rj}u_j + \sum_{j \notin \overline{J}(r), a_{rj} < 0} a_{rj}\ell_j$. Suppose there exists some $s \in \overline{J}(r)$ such that $c_s^k/a_{rs} \leq c_\gamma^k/a_{r\gamma}$ for all $\gamma \in \overline{J}(r) \setminus \{s\}$. If $a_{rs}(u_s - \ell_s) \leq b_r - \overline{U}_r$ then $X_E \subseteq \{x \colon x_s = u_s\}$.

Proof. Note that much of this proof is taken directly from the proof of Theorem 1 in [33]. Let x be an efficient solution with $x_s < u_s$. If $x_j = \ell_j$ for all $j \in \overline{J}(r) \setminus \{s\}$, then a new solution x' constructed from x by setting x'_s to u_s is feasible because

$$\sum_j a_{rj}x'_j = \sum_{j \neq s} a_{rj}x'_j + a_{rs}u_s \leq \tilde{U}_r + a_{rs}(u_s - \ell_s) \leq b_r.$$

Additionally, the value of every objective function improves because $c_s^k < 0$ for all k. This contradicts our assumption of x being efficient. Hence, there exists a $j \in \overline{J}(r) \setminus \{s\}$ with $x_j > \ell_j$. In this case we can construct a new solution x^* from x by decreasing the value of x_j to x'_j while at the same time increasing the value of x_s so that $A_{r\bullet}x^* = A_{r\bullet}x$. In particular, $a_{rs}(x_s^* - x_s) = a_{rj}(x_j - x_j^*)$ holds. The change of objective k can be estimated by

$$\begin{aligned}
c_s^k x_s^* + c_j^k x_j^* &= c_s^k x_s + c_j^k x_j + c_s^k(x_s^* - x_s) - c_j^k(x_j - x_j^*) \\
&= c_s^k x_s + c_j^k x_j + c_s^k \frac{a_{rs}}{a_{rs}}(x_s^* - x_s) - c_j^k \frac{a_{rj}}{a_{rj}}(x_j - x_j^*) \\
&\leq c_s^k x_s + c_j^k x_j + c_s^k \frac{a_{rs}}{a_{rs}}(x_s^* - x_s) - c_s^k \frac{a_{rj}}{a_{rs}}(x_j - x_j^*) \\
&= c_s^k x_s + c_j^k x_j + \frac{c_s^k}{a_{rs}} \big(a_{rs}(x_s^* - x_s) - a_{rj}(x_j - x_j^*)\big) \\
&= c_s^k x_s + c_j^k x_j.
\end{aligned}$$

If $x_s^* = u_s$, the result of the proposition holds. Otherwise, $x_j^* = \ell_j$ holds. Applying this argument iteratively results in an optimal solution with $x_s^* = u_s$ or $x_j^* = j$ for all $j \in \overline{J}(r) \setminus \{s\}$. But as shown before, the latter case contradicts the efficiency of x^*. □

Note that a similar procedure can be followed for the case in which $a_{rj} < 0$ and $c_j^k > 0$ for all k. Now, given two variables x_i and x_j, either both integer or both continuous, we say that x_j *dominates* x_i if (i) $c_j^k \leq c_i^k$ for all k, and (ii) $a_{rj} \leq a_{ri}$ for every r. Note that this variable domination has no relationship with the idea of domination between bound sets. Observe the following lemma, which is an obvious extension from Lemma 1 of [33].

Lemma 3.4. Let x be a feasible solution for an instance of BOMIP and $x_j \succ x_i$. Given $0 < \alpha \in \mathbb{R}$, we define x^* so that

$$x_\gamma^* = \begin{cases} x_\gamma + \alpha & \gamma = i \\ x_\gamma - \alpha & \gamma = j \\ x_\gamma & \text{otherwise.} \end{cases}$$

If $x_j^* = x_j + \alpha \leq u_j$ and $x_i^* = x_i - \alpha \geq \ell_i$, then x^* is feasible and $f_k(x^*) \leq f_k(x)$ for all k.

Proposition 3.5 (Dominating columns)**.** Suppose that x_j dominates x_i in the BOMIP. Then $X_E \subseteq \{x \colon x_j = u_j\} \cup \{x \colon x_i = l_i\}$.

Proof. Again, much of this proof is taken directly from [33]. Let x be an efficient solution such that $x_j < u_j$ and $x_i > \ell_i$. We construct a feasible solution x^* by defining $\alpha = \min\{x_i - \ell_i, u_j - x_j\}$ and applying Lemma 3.4. Since x is efficient and $f_k(x^*) \leq f_k(x)$ for all k, x^* is also efficient. By definition of α, we also have $x_j^* = u_j$ or $x_i^* = \ell_i$. □

Note that one may use the disjunction resulting from Proposition 3.5 to generate valid cutting planes for X_I prior to the start of BB. Additionally, there are also ways to further utilize the structure of dominating columns in order to strengthen variable bounds as described in [33, Theorem 3, Corollary 1 and 2]. These methods for strengthening bounds also extend to the multiobjective case. However, we did not find these methods to be advantageous in practice. Thus, since the description of these additional strategies is quite lengthy, we omit them from this work.

3.4.1.2 Preprocessing

As in the single objective case, the efficiency of BB can be significantly improved if quality solutions can be generated prior to the start of BB. For biobjective problems, this can be accomplished by either (i) using a heuristic method such as that of Soylu [86], or (ii) solving a series of single objective MIPs obtained through the use of a scalarizing technique such as the weighted-sum or ϵ-constraint method [27].

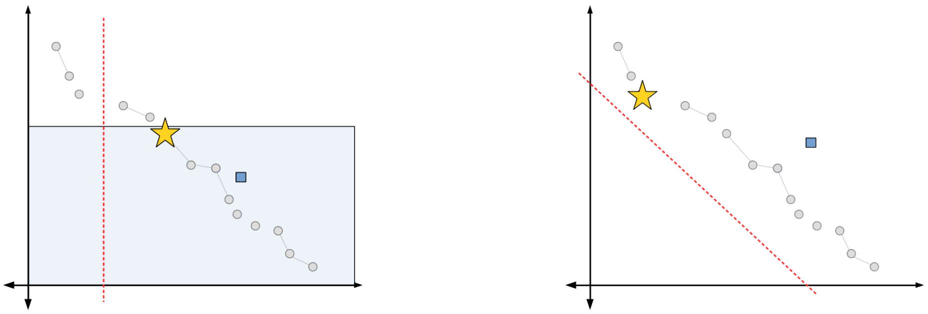
We utilize two different Preprocessing techiniques. Both techniques solve single objective MIPs, subject to a certain time limitation; the first using the ϵ-constraint method, and the second using the weighted-sum approach. We now discuss the various benefits and drawbacks of using either the ϵ-constraint or weighted-sum approaches.

ϵ-constraint: It is well known that for a BOMIP every $y \in Y_N$ can be obtained using the ϵ-constraint method. Unfortunately though, when a MIP formulated using the ϵ-constraint method is not solved to optimality, there are two major drawbacks: (i) each $y \in Y_I$ discovered while processing the MIP must lie within a restricted region of $\mathcal{OS}$, and (ii) the information associated with the best dual bound cannot be utilized.

weighted-sum: The major drawback of the weighted sum method is that when a MIP is formulated using this method, only *supported* Pareto solutions can be found, i.e., those lying on the convex hull of Y_N. There are, however, the following two benefits: (i) $y \in Y_I$ discovered during the MIP solve are not restricted to any particular region of $\mathcal{OS}$, and (ii) the best dual bound is valid for all $y \in Y_I$ and can therefore be used to create a cutting plane in $\mathcal{OS}$.

As can be seen, there is a certain level of trade-off present between the ϵ-constraint method and the weighted sum method. The pros and cons of each technique are illustrated in Figures 3.3a and 3.3b. For each of these figures, we have the following: (i) Y_N, which we assume to be unknown, is shown in grey, (ii) the optimal solution, which we assume is not known at termination of the MIP solve, is depicted as a yellow star, (iii) the best known solution at termination is shown as a blue square, and (iv) the level curve associated with the best known dual bound at termination is shown as a dotted red line. Note that for Figure 3.3a, we assume that ϵ is defined so that the feasible region is restricted to the light blue box.

We now present Algorithms 3.5 and 3.6 in which we describe our proposed ϵ-constraint and weighted sum based preprocessing procedures. On line 3 of Algorithm 3.5 we solve the MIP associated with f_λ. Recall that λ is computed so that the level curves of f_λ have the same slope as the line segment joining y_I^1 and y_I^2. On line 5 we then use the solution of this MIP to compute horizontal and vertical step sizes, h_1 and h_2. These step sizes are then used to sequentially increase the values of ϵ_1 and ϵ_2 which are used on line 7 to construct new MIPs, using the ϵ-constraint problem, which may yield new, undiscovered Pareto solutions. On lines 8 and 9 we modify the step sizes h_1 and h_2. If the MIP solved on line 7 yields a new, previously undiscovered Pareto solution, we decrease



(a) ϵ-constraint method

(b) Weighted sum method

Figure 3.3: Bound information when a single objective MIP terminates early

Algorithm 3.5 Preprocessing based on the ϵ-constraint method.
Input: y_I^1, y_I^2 and a nonnegative value for parameter ρ.
Output: An initialized set of Pareto solutions $\mathcal{N}_0 \subseteq Y_N$.

```
function PreprocessingMethod1($y_I^1, y_I^2, \rho$)
    Let $\mathcal{N}_0 = \emptyset$.
    Solve the MIP $\min\{f_\lambda(x) : x \in X_I\}$ to obtain $y_I^\lambda \in Y_I$.
    Add a cutting plane to $X$ which lies on the level curve of $f_\lambda$ associated with the best found dual solution.
    Set $h_1 = \frac{(y_I^2)_1 - (y_I^\lambda)_1}{60}$, $\epsilon_1 = (y_I^\lambda)_1 + h_1$, $h_2 = \frac{(y_I^1)_2 - (y_I^\lambda)_2}{60}$ and $\epsilon_2 = (y_I^\lambda)_2 + h_2$.
    for $k \in \{1,2\}$ do
        while $\epsilon_k > (y_I^k)_k$ do solve the MIP $P_k(\epsilon_k) := \min\{f_{\{1,2\}\setminus\{k\}}(x) : x \in X_I, f_k(x) \leq \epsilon_k\}$ to obtain $y^* \in Y_N$.
            if $\mathcal{N}_0 \not\succ y^*$ then set $h_k = \frac{h_k}{1+\rho}$.
            else set $h_k = \max(5-\rho, 1)h_k$.
            for each $x \in X_I$ found while solving $P_k(\epsilon_k)$ do let $N =$ GenerateDualBd($s(x)$) and set $\mathcal{N}_0 = (\mathcal{N}_0 \cup N)^{nd}$.
            Set $\epsilon_k = \epsilon_k + h_k$.
    Return $\mathcal{N}_0$.
```

the step size. Otherwise we increase it. This allows us the continue searching for additional new solutions in locations of $\mathcal{OS}$ which are near previously discovered solutions, and to cease searching in areas in which new solutions are not being generated. Note that the amount in which the step sizes are increased or decreased depends on the value of the parameter ρ. Also note that each time we solve a MIP, we utilize its solution to update $\mathcal{N}_s$.

In Algorithm 3.6 we compute several sets of weights which we utilize in the weighted-sum approach to generate Pareto solutions. We initialize the set of weights Λ on line 3 with the weight λ for which the level curves of f_λ have the same slope as the line segment joining y_I^1 and y_I^2. We use σ to represent the number of weights for which MIPs will be solved in a given iteration. We deem an iteration successful if at least a fifth of the solved MIPs reveal previously undiscovered

Algorithm 3.6 Preprocessing based on the weighted-sum method.
Input: A nonnegative value for parameter ρ.
Output: An initialized set of Pareto solutions $\mathcal{N}_0 \subseteq Y_N$.

```
1:  function PreprocessingMethod2($\rho$)
2:      Let $\mathcal{N}_0 = \emptyset$.
3:      Set $\Lambda = \{\lambda\}$, $\Lambda' = \{0, 1\}$ and $t = 0$.
4:      while $t \leq \rho$ do
5:          Set $\tau = 0$ and $\sigma = |\Lambda|$.
6:          for $\lambda' \in \Lambda$ do remove $\lambda'$ from $\Lambda$ and add it to $\Lambda'$.    (Assume $\Lambda'$ is always sorted in increasing order.)
7:              Solve the MIP $P(\lambda') := \min\{f_{\lambda'}(x) : x \in X_I\}$ to obtain $y^{\lambda'} \in Y_I$.
8:              Add a cutting plane to $X$ which lies on the level curve of $f_{\lambda'}$ associated with the best found dual solution.
9:              if $\mathcal{N}_0 \not\succ y^{\lambda'}$ then set $\tau = \tau + 1$.
10:             for each $x \in X_I$ found while solving $P(\lambda')$ do let $N =$ GenerateDualBd($s(x)$) and set $\mathcal{N}_0 = (\mathcal{N}_0 \cup N)^{nd}$.
11:         for each adjacent pair $(\lambda_1, \lambda_2) \in \Lambda'$ do add $\frac{\lambda_1+\lambda_2}{2}$ to $\Lambda$.
12:         if $\tau < \frac{\sigma}{5}$ then set $t = t + 1$.
13:     Return $\mathcal{N}_0$.
```

Pareto solutions. We use τ to count the number of unsuccessful iterations. On line 11 we increase the number of weights that will be used in the next iteration by computing the next set of weights so that it contains the midpoint of each pair of adjacent weights in the set Λ', which is the set of previously used weights together with 0 and 1. The process then terminates when the number of unsuccessful iterations exceeds the value of the parameter ρ. As we did with Algorithm 3.5, we also utilize the solution of each MIP we solve in this procedure to update $\mathcal{N}_s$.

3.4.1.3 Probing

After Preprocessing, a probing technique can be used to strengthen the bounds on each integer variable, as stated below.

Proposition 3.6 (Probing on x_i)**.** Let x_i be an integer variable. Fix $x_i = l_i$ and solve the BOLP relaxation. Let $\mathcal{L}_{l_i}$ be the Pareto set of this MOLP. If $\mathcal{U}_0 \succ \mathcal{L}_{l_i}$ then $X_E \subseteq \{x \colon x_i \geq l_i + 1\}$.

Proof. Recognize that $\mathcal{L}_{l_i}$ dominates every $y_I \in Y_I$ for which $y_I = \mathbf{f}(x)$ and $x_i = l_i$. The desired result follows since $\mathcal{U}_0 \succ \mathcal{L}_{l_i}$. □

This procedure can be repeated multiple times for a given integer x_i and then iterated over each additional integer variable x_j. Furthermore, a similar procedure to that of Proposition 3.6 exists for tightening the upper bound. We point out that there are likely many more tasks that could be performed during Presolve and/or Preprocessing that could further impact the performance of BB. However, our goal here is not to develop extensive procedures for these tasks, but to put together an initial implementation that highlights some of what can be done.

3.4.2 Additional Notes on Node processing

Recall that we discussed the major aspects of node processing in Section 3.3. Here we discuss a few additional, though non-essential, tasks that we perform while processing a node.

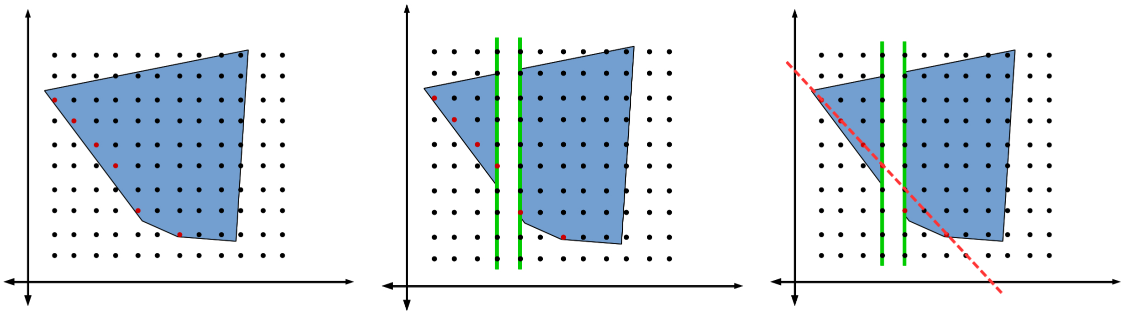
3.4.2.1 Objective Space Fathoming

After processing a node, we perform an additional type of fathoming which we refer to as *objective-space fathoming*. After updating $\mathcal{N}_s$, we impose bounds on f_1 and f_2 which "cut off" portions of $\mathcal{OS}$ in which we have found that $\mathcal{U}_s \succ (Y_s)_I$. In some cases the remaining subset of $\mathcal{OS}$ consists of disjoint regions. When this occurs, we implement objective-space fathoming by branching on f_1 and f_2 bounds which generate the desired disjunctions in $\mathcal{OS}$. In these cases, objective-space fathoming resembles the "Pareto branching" of [90] and "objective branching" of [71].

3.4.2.2 Bound Tightening

In order to increase the likelihood of fathoming, we utilize a few different strategies for tightening the bound $\mathcal{L}_s$. The first strategy we use is the generation of locally valid cutting planes. We do this in two ways: (i) we generate discjuntive cuts based on disjunctions observed in $\mathcal{OS}$ when performing $\mathcal{OS}$ fathoming, and (ii) we convert the BOLP relaxation associated with s to a BOMIP, allow CPLEX to process its root node, and add all cuts generated by CPLEX for this BOMIP to s as local cuts.

It is widely accepted that for single objective MIPs, locally valid cutting planes are not particularly helpful for improving the performance of BB. However, locally valid cutting planes can have a significantly greater impact on BOMIPs. To see this, observe Figure 3.4. Assume that Figure 3.4a displays an instance of BOMIP for which the (f_1, f_2)-space and the X-space are one and the same, i.e., this instance contains only two variables y_1 and y_2, both integer, and $f_1 = y_1$ and $f_2 = y_2$. The constraints of this instance yield the blue polytope, and the integer lattice is indicated by the black dots. The red dots represent the Pareto-optimal solutions. Suppose that branching is performed as shown in Figure 3.4b. Notice that all Pareto optimal solutions in the left branch can be revealed by a single locally valid cutting plane, as shown by the red dashed line in Figure 3.4c. Also notice that this could never be accomplished through the use of globally valid cuts.



(a) Example instance of BOMIP (b) After branching (c) Locally valid cut

Figure 3.4: An example showing the usefulness of locally valid cuts for BOMIP

3.4.3 Branching

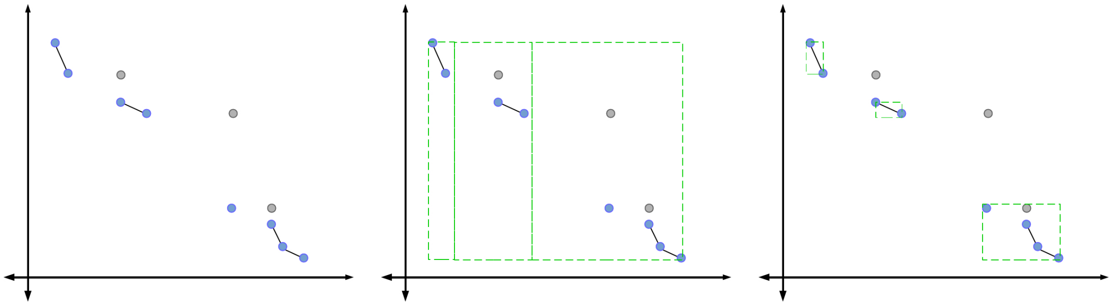
In general, any rule for selecting a branching variable is permissible. However, it should be noted that for BOMIP several $y \in Y$, and consequently several $x \in X$, may be discovered while processing a node s. In fact, our implementation requires solving at least three LPs at each node. Since the variables may take on different values at each solution, it is possible that an integer variable takes a fractional value at some of these solutions and not at others. Because of this, we use a scoring scheme for branching in which each integer variable is given a score. Of the variables with the highest score, the one with the highest index is selected for branching. The score of x_i is increased if: (i) x_i is fractional at the LP solution associated with objective f^k, $k \in \{1, 2, \lambda^s\}$, (ii) x_i changes value at a pivoting step of Algorithm 3.2, or (iii) multiple single objective MIPs are solved to optimality at s and x_i takes different values for at least two of the MIP solutions.

After a branching decision has been made we utilize probing, as introduced in Proposition 3.6, to strengthen bounds on each variable for both of the resulting subproblems. We do this for several reasons: (i) we may find during this process that our branching decision results in an infeasible subproblem, in which case we can discard the infeasible subproblem, enforce that the variable bounds associated with the feasible subproblem be satisfied at any child node of s, and choose a new branching variable; (ii) because much work in biobjective BB is dedicated to fathoming, we want to generate the strongest dual bound possible, which probing helps us to do; (iii) since processing a node in biobjective BB is an expensive operation, we seek to limit the number of nodes explored and probing aids in this endeavor by reducing the number of possible future branching decisions. We found during testing that this probing scheme at each node was extremely powerful, both in reducing the number of nodes processed during BB as well as overall running time. See Table 3.1 in Section 3.5 for evidence of this.

3.4.4 Additional Improvements

3.4.4.1 Exploiting gaps in $\mathcal{OS}$

Due to the noncontinuous, nonconvex nature of the Pareto set of a BOMIP, there are occasionally large gaps between Pareto solutions in $\mathcal{OS}$. If this occurs, the likelihood that $\mathcal{L}_s \subseteq \mathcal{U}_s$ is significantly decreased for each node. Hence, this can result in an extreme amount of computational effort which yields no additional Pareto solutions. One way to combat this issue is to observe the solutions obtained during Preprocessing and record locations in $\mathcal{OS}$ where large gaps exist between discovered solutions. One can then split $\mathcal{OS}$ into a series of subregions based on the locations of these gaps and solve single objective MIPs (using objectives f_1 and f_2) within each subregion in order to remove locations containing no Pareto solutions. Afterwards BB can be run in each subregion rather than over the entire $\mathcal{OS}$. To aid in understanding this idea, observe Figure 3.5. Here Pareto solutions are shown in blue and subregions in $\mathcal{OS}$ are indicated by green dashed lines.



(a) Gaps

(b) Slitting $\mathcal{OS}$

(c) Reducing the subregions

Figure 3.5: Large gaps between solutions in $\mathcal{OS}$

3.4.4.2 Measuring Performance

In single objective BB one can terminate the procedure at any time and obtain a measure of the quality of the best known solution in terms of the gap between this solution and the best known dual bound. We propose a similar scheme for biobjective BB. Let $\mathscr{O}_{s^*}$ represent the set of open nodes after a node s^* has been processed. After processing s^* the global dual bound, denoted $\mathcal{DB}_{s^*}$, is the nondominated subset of $(\cup_{s \in \mathscr{O}_{s^*}} \mathcal{L}_s)$. Therefore, if BB is terminated after s^* is processed, the performance of BB can be quantified by measuring the distance between $\mathcal{DB}_{s^*}$ and $\mathcal{U}_{s^*}$. A natural metric to use for measuring this distance is the Hausdorff metric: $d_H(\mathcal{DB}_{s^*}, \mathcal{U}_{s^*}) := \max\{\sup_{i \in \mathcal{DB}_{s^*}} \inf_{j \in \mathcal{U}_{s^*}} d(i,j), \sup_{j \in \mathcal{U}_{s^*}} \inf_{i \in \mathcal{DB}_{s^*}} d(i,j)\}$. Unfortunately the nonconvex nature of $\mathcal{U}_s$ makes the Hausdorff metric difficult to use since it cannot be computed using a linear program. In our implementation $\mathcal{U}_{s^*}$ is stored as the individual line segments and singletons comprising $\mathcal{N}_{s^*}$ using the data structure of [3]. $\mathcal{DB}_{s^*}$ is computed by generating the points and line segments comprising its nondominated subset, which are also stored using the data structure of [3]. Thus, rather than explicitly computing $d_H(\mathcal{DB}_{s^*}, \mathcal{U}_{s^*})$, we instead compute

$$\mathcal{G}_{s^*} := \max\{d_H(\mathcal{DB}_{s^*}, \mathcal{S} + \mathbb{R}^2_{\geq}) : \mathcal{S} \in \mathcal{N}_{s^*}\} \tag{3.8}$$

via pairwise comparison of the points and line segments comprising $\mathcal{DB}_{s^*}$ and $\mathcal{N}_{s^*}$. Note, though, that $\mathcal{G}_{s^*}$ is a clear upper bound on $d_H(\mathcal{DB}_{s^*}, \mathcal{U}_{s^*})$. Recognize, though, that $\mathcal{G}_{s^*}$ is an absolute measurement and so it is difficult to use to compare the performance of BB on multiple instances of BOMIP. Thus, in practice we use a percentage calculated as

$$\overline{\mathcal{G}_{s^*}} := 100 \frac{\left|\max\{y_1^2 - y_1^1, y_2^1 - y_2^2\} - \mathcal{G}_{s^*}\right|}{\max\{y_1^2 - y_1^1, y_2^1 - y_2^2\}}. \tag{3.9}$$

3.4.5 The BB Algorithm for BOMIP

We now provide Algorithm 3.7 which contains an outline of the BB procedure we propose for BOMIP. In the next section we conduct a set of computational experiments designed to test the practical performance of the presented BB scheme.

3.5 Experimental Results

We implemented our BB scheme using the C programming language and the CPLEX optimization package (version 12.6 [45]). Boland et al. [16] graciously shared their code with us and so we were able to compare the performance of our BB with the triangle splitting method, which

Algorithm 3.7 Perform BB to obtain the Pareto set of an instance of BOMIP.
Input: An instance $\mathcal{I}$ of BOMIP.
Output: The Pareto set of instance $\mathcal{I}$.

```
1:  function BBsolve($\mathcal{I}$)
2:      Set $\mathscr{L} = \emptyset$.
3:      Simplify $\mathcal{I}$ by performing standard primal presolve and biobjective duality fixing and exploitation of singleton and dominating columns.
4:      for $k \in \{1, 2\}$ do solve the MIP $\min\{f_k(x) : x \in X_I\}$ to obtain $y_I^k \in Y_I$.
5:      Select $\rho \geq 0$ and run either PreprocessingMethod1($y_I^1, y_I^2, \rho$) or PreprocessingMethod2($y_I^1, y_I^2, \rho$) to return $\mathcal{N}_0$.
6:      Perform probing to further simplify the structure of $\mathcal{I}$.
7:      Add the continuous relaxation of $\mathcal{I}$ to $\mathscr{L}$.
8:      while $\mathscr{L} \neq \emptyset$ do select $s$ from $\mathscr{L}$.
9:          Run ProcessNode($s$).
10:         if $s$ is not fathomed then perform $\mathcal{OS}$ fathoming.
11:             if the remaining nondominated portion of $\mathcal{OS}$ consists of disjoint regions then perform Pareto branching on an $\mathcal{OS}$ disjunction. Add the resulting subproblems to $\mathscr{L}$.
12:             else select the variable with highest score for branching.
13:                 Perform probing to simplify the structure of each of the subproblems resulting from the current branching decision.
14:                 if probing reveals an infeasible subproblem then impose the restrictions of the feasible subproblem and select the variable with the next highest score for branching. Repeat Line 13.
15:                 else branch on the selected variable. Add the resulting subproblems to $\mathscr{L}$.
16:     Return $\mathcal{N}_{s^*}$, where $s^*$ is the last node for which ProcessNode was called.
```

we recall is a search method in the objective space. In preliminary tests we also compared with the BB method of [11]. However, their implementation was not complete and so the results we obtained were not comparable. All testing was conducted using the Clemson University Palmetto Cluster. Specifically, we used an HP SL250s server node with a single Intel E5-2665 CPU core with 32GB of RAM running Scientific Linux 6.4.

Our initial test set consisted of the instances examined in [11, 16]. The instances from [11] contained either 60 variables and 60 constraints, or 80 variables and 80 constraints. From here we refer to these instances as “Belotti60” and “Belotti80.” We label the instances from [16] in a similar way. This instances they consider in the journal version of their paper are labelled “Boland80,” “Boland160,” and “Boland320” (we do not solve instances with less than 60 constraints or variables). We also utilize instances that were considered in a previous version of this paper [15]. To maintain constancy with the way these instances were labelled in [15], we refer to these instances as “Boland16,” “Boland25,” and “Boland50,” although the respective total number of variables and constraints for each of these instance sets is approximately 800, 1250 and 2500. Due to our success on

Table 3.1: Experiment 1 – Measuring the impact of presolve techniques.

		All Off			Duality Fixing On			Singleton Columns On			Dominating Columns On		
Instance	#	Time	Nodes	$\overline{\mathcal{G}_{s*}}$	Time	Nodes	$\overline{\mathcal{G}_{s*}}$	Time	Nodes	$\overline{\mathcal{G}_{s*}}$	Time	Nodes	$\overline{\mathcal{G}_{s*}}$
Belotti60	30	7	76	54	7	76	54	7	76	54	7	76	54
Belotti80	30	16	87	56	17	87	56	17	87	56	16	87	56
Boland80	5	26	541	38	24	493	33	26	541	38	28	541	38
Boland160	5	899	2,873	17	808	2,801	15	893	2,873	17	895	2,873	17
Boland320⊛	5	31,822	14,262	9	32,897	17,628	41	31,841	14,262	9	31,589	14,262	9
Boland16	4	9	97	18	8	96	18	9	97	18	9	97	18
Boland25	4	61	327	15	54	338	15	60	327	15	61	327	15
Boland50	4	2,343	2,531	19	1,461	2,084	19	2,395	2,531	19	2,323	2,531	19

⊛ – Only 4 of 5 instances completed.

these instances we felt the need to create a more difficult test set. Hence we also tested on biobjective variants of some instances from MIPlib 2010 [54] – we chose only those instances that were marked easy, are mixed-integer and not pure integer and were relatively small in size (up to approximately 200 integer variables). For each instance, we generated six secondary objective functions using a mix of randomized and deterministic procedures with the hope that there was some conflict in the two objectives. We discarded instances for which: (i) the Pareto set was a singleton, or (ii) the second objective was unbounded, or (iii) the MIP associated with either f_1 or f_2 took over 12 hours to solve. We set a maximum solution time of 12 hours for all instances.

We began our tests by turning off all nonessential features of our BB procedure, and then sequentially turning on various features to test their impact on the overall procedure. If a particular feature of our BB procedure was deemed effective in reducing the overall effort required to solve instances of BOMILP, this feature was left on for the remainder of the tests, otherwise it was turned back off. We first test the utility of the various presolve procedures discussed in Section 3.4.1.

3.5.1 Presolve Techniques

Table 3.1 contains the results of our first computational experiment. We report the average computation time in seconds to solve instances of each type, the average number of nodes explored, and the average duality gap percentage computed after processing the root node. Note that in for this test we utilized ProprocesingMethod2 with ρ set to zero.

Notice from Table 3.1 that the results for duality fixing show the opposite pattern for the Boland320 instances than for all other instances. This is because, for an unknown reason, fixing several variables during presolve had a negative impact on preprocessing. This caused fewer solutions to be discovered during preprocessing and thus had an overall negative impact on the rest of the BB procedure. We felt, though, that the positive impact duality fixing had on other instance sets warranted leaving this feature on for the remainder of our tests. Also observe from Table 3.1 that the exploitation of neither singleton nor dominating columns had an impact on the overall BB procedure. This was primarily due to the fact that there were very few occurrences of either of these types of columns. We opted to turn off the exploitation of singleton columns for the remainder of our tests, but we left on the exploitation of dominating columns. Our reasoning was that singleton columns have no impact on BB that extends beyond presolve, while dominating columns result in disjunctions from which we can generate global cutting planes. Hence, we left on the exploitation of dominating columns in order to test the impact of generating these cuts in later tests.

3.5.2 Preprocessing

In our next test we examined the impact of the two preprocessing techniques discussed in Section 3.4.1, as well as a hybrid method we derived as a combination of the two presented procedures. In our initial implementation of this test we used each of these three methods with rho assigned each integer value in $[0, 5]$. Recognize from Algorithms 3.5 and 3.6 that each of the proposed preprocessing procedures are designed so that the total number of Pareto solutions computed should have a positive correlation with the value of ρ. We determined that ProprocesingMethod1 performed poorly for $\rho \leq 1$ and ProprocesingMethod2 performed poorly for $\rho \geq 2$. We also discovered that the impact of ρ on overall solution time varied with the size of the instance solved. As a result, we also implemented modified preprocessing procedures in which the value of ρ is automatically computed as a function of the size of an instance. Figures 3.6 and 3.7 respectively contain performance profiles of CPU time for instances of size 80 and smaller, and size greater than 80. We note that in the legends for these profiles we use "e," "w," and "hy" to denote ProprocessingMethod1 (based on the ϵ-constraint method), PreprocessingMethod2 (based on the weighted sum approach), and the hybrid method. The subsequent numbers indicate the value of ρ. Additionally, the "term" vary indicates that ρ was automatically computed as a function of instance size.

Observe from Figures 3.6 and 3.7 that the hybrid preprocessing approach did not perform well compared to the other approaches. Now consider ProprocessingMethod2. Although variants of this procedure performed well for smaller instances, the same is not true for larger instances. PreprocessingMethod1, on the other hand, performed quite well on all instances. Notice, though, that values of ρ near two performed quite well for small instances while values near five performed

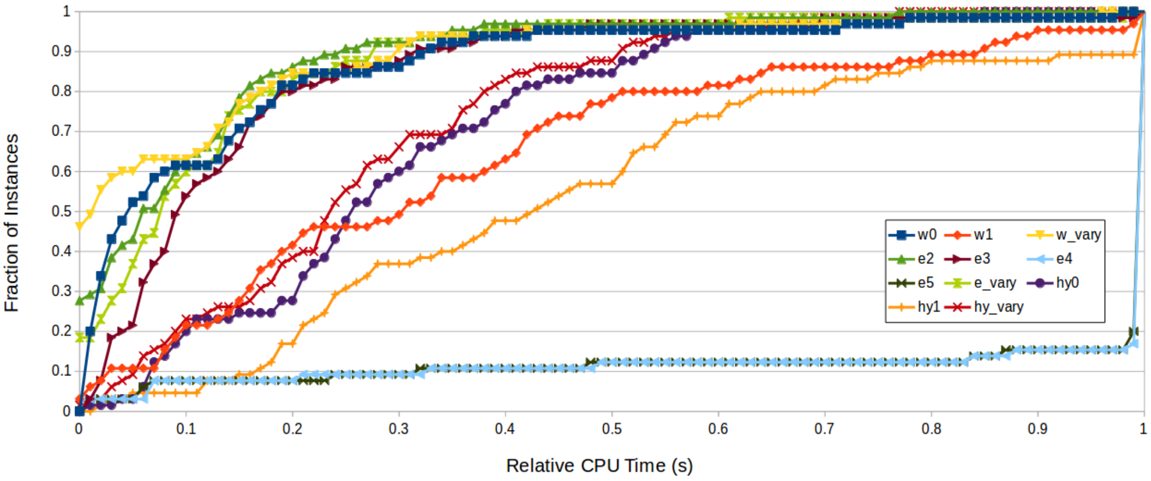


Figure 3.6: Performance profile of CPU time for instances of size 80 and less.

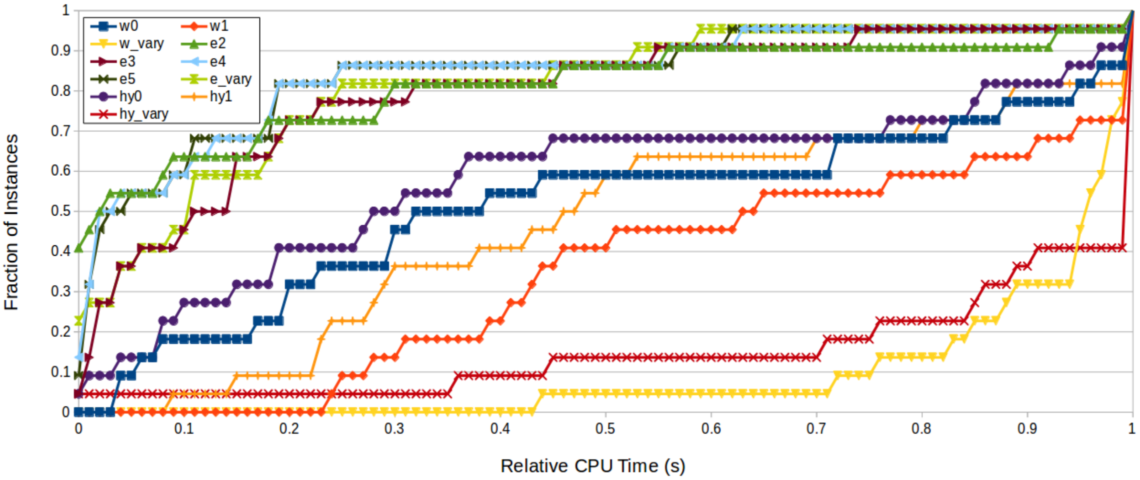


Figure 3.7: Performance profile of CPU time for instances of size greater than 80.

extremely poorly. However, for larger instances values of ρ near five seem to outperform almost every other procedure. Due to the consistent performance of the variant of PreprocessingMethod1 in which the value of ρ was computed automatically as a function of instance size, we opted to use this approach for the remainder of our tests.

3.5.3 Probing and Pareto Branching

The next test we performed was designed to examine the utility of the variable probing procedure discussed in Section 3.4.1, both used directly after preprocessing and at each node prior to branching, and the Pareto branching that we perform when $\mathcal{OS}$ fathoming, described in Section 3.4.2, results in disjoint feasible regions of $\mathcal{OS}$. The results of this experiment are given in Table 3.2.

Table 3.2: Experiment 3 – Measuring the impact of probing and Pareto branching.

	Total	All Off			Initial Probing On			Probing During Branching On		Pareto Branching On	
Instance	Num	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	Time	Nodes
Belotti60	30	7	73	54	8	78	54	5	47	7	72
Belotti80	30	21	95	46	21	92	45	13	60	18	86
Boland80	5	19	390	20	19	397	20	10	217	18	407
Boland160	5	667	2,497	25	679	2,506	25	244	978	534	2,569
Boland320	5	19,902	10,209	6	19,160	9,971	6	6,865	3,720	14,348	9,583
Boland16	4	10	83	10	10	84	10	8	62	11	101
Boland25	4	52	394	52	56	427	52	34	290	50	381
Boland50	4	1,204	1,706	16	1,556	2,008	16	987	1,426	1,136	2,174

Observe from Table 3.2 that when utilizing probing directly after preprocessing, in most cases the total CPU time and number of nodes processed increased. However, performing the same probing procedure prior to branching at each node had an extremely positive impact on the overall performance of BB, significantly lowering total CPU time and the number of explored nodes. We also found that Pareto branching had an overall positive impact on BB performance. For the remainder of our tests we opted to cease probing directly after preprocessing, but to still employ probing during branching and Pareto branching.

3.5.4 Local Cuts

The next test we performed was designed to test the utility of various cut generation procedures that we employed. We divided this test into two parts, (a) and (b). In part (a) we examined the performance of BB while applying the local cut generation procedure we discussed in Section 3.4.2, the generation of globally valid cutting planes from disjunctions implied by pairs of dominating columns, and the generation of locally valid cuts from $\mathcal{OS}$ space disjunctions discovered during $\mathcal{OS}$ fathoming. For part (b) of the experiment we decided to test the utility of a new procedure for generating globally valid cuts after preprocessing, but prior to processing the root node. In this procedure we preselect a number of of values of λ, evenly distributed in $(0,1)$, and pass the MIP $\min\{\lambda f_1(x) + (1-\lambda) f_2(x) : x \in X_I\}$ to CPLEX. We allow CPLEX to process the root node of this MIP, afterwards we extract the cutting planes discovered by CPLEX and add them to our original BOMIP as global cuts. The motivation behind this approach is that, because the implementation of our biobjective BB procedure is an adaptation of standard CPLEX single objective BB, modified through the use of callbacks, the standard cut generation procedure of CPLEX will only generate cuts based on the objective associated with the single objective problem we pass to CPLEX. This means that the cuts generated by the default CPLEX cut generation procedure are only useful in closing the duality gap in a small subregion of $\mathcal{OS}$. We designed our procedure to combat this issue. The results of parts (a) and (b) of this experiment are given in Tables 3.3 and 3.4, respectively.

Table 3.3: Experiment 4, part (a) – Measuring the impact of cut generation procedures.

	Total	All Off			Local Cut Generation On			Global Cuts From Dominating Column Disjunctions On			Local Cuts From $\mathcal{OS}$ Disjunctions On	
Instance	Num	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes
Belotti60	30	5	52	44	7	52	44	5	52	44	5	52
Belotti80	30	13	61	45	16	57	45	13	61	45	13	61
Boland80	5	8	219	20	8	215	20	8	219	20	8	219
Boland160	5	189	1,015	21	198	1,050	21	188	1,015	21	190	1,007
Boland320	5	4,417	4,042	5	5,005	4,087	5	4,545	4,103	5	4,471	4,100
Boland16	4	7	60	10	7	60	10	7	60	10	8	61
Boland25	4	29	272	46	29	265	46	29	272	46	32	273
Boland50	4	658	1,437	16	693	1,407	16	644	1,495	16	709	1,528

Observe from Table 3.3 that utilizing each of the displayed methods for cut generation had a negative impact on the CPU time used during BB. A couple of these methods did aid in reducing the number of nodes explored during BB, but not substantially. As a result, we opted to turn off all of these cut generation schemes for the remainder of our tests. There are a couple of important notes to be made concerning cut generation, though. First, it is important to recognize that the potential impact of generating locally valid cuts for BOMIP is likely not properly displayed by the results of this experiment. The primary reason for this is that CPLEX does not allow for the addition of locally valid cutting planes except during the execution of a user-cut-callback. However, such a callback is only employed intermittently and quite rarely once a certain depth of the BB has been reached. This is unfortunate, since it seems that locally valid cuts may have an increasingly significant impact on the reduction of the duality gap as the depth of the BB tree increases. Another important thing to note concerning these cut generation schemes is that are two ways in which we can pass globally valid cuts to CPLEX, and each is limited in its own way. First, we can pass a global cut to CPLEX specifically as a cut. However, when doing so, CPLEX will only utilize this cut if it detects a solution at which this cut is violated. This is unfortunate though, since as we have discussed, CPLEX is only aware of solutions generated from a single objective. Many of the solutions generated during BB are generated by us, during a callback, and not by CPLEX. Thus, even though solutions may be generated which violate a cut we have passed to CPLEX, the cut still never be utilized. The second way we could pass a cut to CPLEX is by explicitly adding it to the

BOMIP model as an additional row. This forces the utilization of this cut, but adding too many cuts in this way causes CPLEX to need to perform a significant amount of additional book-keeping and therefore typically has an overall negative impact on BB.

Table 3.4: Experiment 4, part (b) – Measuring the impact of cut generation procedures.

					Add Extra Global Cuts Using Various λ's Prior to Start of BB															
		Off			# of λ's: 2			3			5			# of λ's: 9			17			
Instance	#	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	
Belotti60	30	5	52	48	6	52	49	6	54	48	6	53	43	7	54	42	9	54	42	
Belotti80	30	13	61	46	14	62	46	14	62	46	15	60	45	17	61	45	21	61	45	
Boland80	5	8	219	20	8	219	20	8	218	14	8	219	9	8	223	6	7	219	6	
Boland160	5	185	1,015	25	185	1,015	23	185	1,015	20	185	1,011	20	181	979	20	181	985	20	
Boland320	5	4,347	4,028	6	4,274	3,986	6	4,257	3,976	6	4,343	3,984	6	4,316	3,946	6	4,216	3,870	6	
Boland16	4	7	60	10	7	60	10	7	60	10	7	60	10	7	60	10	7	60	10	
Boland25	4	29	272	52	28	272	52	29	277	52	26	241	52	26	237	52	23	221	52	
Boland50	4	639	1,470	16	646	1,494	16	655	1,487	16	666	1,505	16	707	1,544	16	587	1,342	16	

Observe from Table 3.4 that there is no set of instances which displays an overall decrease in CPU time as the number of utilized values of λ increases. We note that for the instances from [11] there is an overall increase in running time, while the instances from [16] display a haphazard pattern, increasing on some occasions and decreasing on others. The reason for the pattern displayed by the instances from [11] is that, although several cutting planes were generated for each used value of λ, as we described in our dicussion of Table 3.3, in order for these cuts to be utilized by CPLEX we were forced to add them as rows to the BOMIP model, which caused a significant increase in the computational overhead. The reason for the pattern displayed by the instances from [16] is that for the majority of these instances the single objective MIP associated with each value of λ was solved by CPLEX before any cutting planes were generated. Thus, there were rarely cuts to be extracted and copied. The variation in running times and number of nodes processed seems to be due to a difference in the order in which CPLEX processed nodes during the biobjective BB procedure. As this procedure of generating additional cutting planes did not result in a decrease in CPU time spent in BB, we opted to turn off this procedure for the remainder of our tests.

3.5.5 Additional Improvements

For our next experiment we decided to test potential simplifications to Fathoming Rule 3 and the generation of $\mathcal{L}_s^{nd}$. We now describe these two improvements, beginning with that of Fathoming Rule 3. Recognize from Algorithm 3.3 that if we have a node s for which $\mathcal{U}_s \succ \mathcal{L}_s$, but Fathoming Rules 1a and 2a fail, Fathoming Rule 3 does not cease until every defining line segment of $\mathcal{L}_s^{nd}$ is generated. To attempt to reduce the time spent executing Fathoming Rule 3 on these occasions, we implemented the following procedure:

1. Select $\alpha \in \mathbb{Z}_+$.

2. After α lines segments have been generated during the execution of Algorithm 3.3, for each newly generated line segment dominated by U_s, extend the line segment so that the first component of its left-most point is $(y_s^1)_1$. If this extended line segment is dominated by $\mathcal{U}_s$, then $\mathcal{L}_s$ is also dominated so fathom node s.

An example of this procedure is depicted in Figure 3.8. In Figure 3.8 $\mathcal{N}_s$ is shown in blue and $\mathcal{L}_s^{nd}$ is shown in red. We assume the two right-most segments of $\mathcal{L}_s^{nd}$ have already been generated and shown to be dominated by $\mathcal{U}_s$. Hence, we can see that the node being considered can be fathomed after the generation of the "extended segment" without needing to generate the final segment of $\mathcal{L}_s^{nd}$.

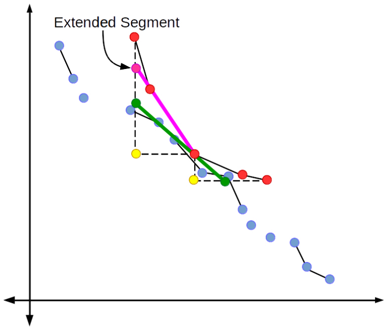


Figure 3.8: Simplification of Fathoming Rule 3.

We now consider the simplification of the generation of $\mathcal{L}_s^{nd}$. In order to simplify this procedure we cease generating segments in $\mathcal{L}_s^{nd}$ if any segment in generated which is dominated by $\mathcal{U}_s$. The results we obtained from this experiment are shown in Table 3.5.

Note that in Table 3.5 we do not report the number of nodes processed when the simplified version of Fathoming Rule 3 is employed because there is no change in the number of nodes processed using this method and the original implementation. Unfortunately, neither of our proposed

Table 3.5: Experiment 5 – Improvements to Fathoming Rule 3 and the generation of $\mathcal{L}_s^{nd}$.

		Both		Check For Early Termination of Fathoming Rule 3						Check For Early Termination of	
	Total	Off		$\alpha = 0$	5	10	15	20	25	$\mathcal{L}_s^{nd}$ Generation	
Instance	Num	Time	Nodes	Time	Time	Time	Time	Time	Time	Time	Nodes
Belotti60	30	5	52	5	5	5	5	5	5	5	52
Belotti80	30	13	61	13	13	13	13	13	13	13	60
Boland80	5	8	218	8	8	8	8	8	8	8	218
Boland160	5	196	1,015	195	194	192	194	196	194	199	1,045
Boland320	5	4,384	4,094	4,487	4,383	4,382	4,469	4,395	4,399	4,380	4,117
Boland16	4	7	60	7	7	7	7	7	7	7	61
Boland25	4	28	272	28	28	28	28	28	28	28	261
Boland50	4	651	1,477	645	678	673	638	654	629	638	1,468

simplifications resulted in improved CPU times for BB, so we turned off these simplifications for the remainder of our tests.

3.5.6 Exploiting $\mathcal{OS}$ Gaps and Comparing with Triangle Splitting

We have now presented the results of all experiments designed to study the impact of the various aspects of our BB procedure. We now present the results of an experiment designed to test the performance of our BB against that of the triangle splitting method of [16]. For this experiment we solved all the same instances we used in our previous tests and employed two variants of our BB procedure, one in which we utilized the $\mathcal{OS}$ splitting procedure we discussed in Section 3.4.4.1 and one in which we utilized our standard implementation. We compared our results with that of the triangle splitting method of [16]. The results of this test are given in Table 3.6.

Table 3.6: Experiment 6 – Comparison with the triangle splitting method.

	Total	Standard BB		BB with Exploiting $\mathcal{OS}$ Gaps			Triangle Splitting
				Time			
Instance	Num	Time	Nodes	Total	Parallel†	Nodes	Time
Belotti60	30	5	52	6	3	33	16
Belotti80	30	13	61	15	7	42	37
Boland80	5	8	218	7	6	203	68
Boland160	5	185	1,015	167	162	954	661
Boland320	5	4,433	4,107	4,501	4,501	4,200	8,620
Boland16	4	7	60	7	7	60	14
Boland25	4	28	272	26	26	284	68
Boland50	4	664	1,488	623	440	1,354	631

† – Presolve/Preprocessing time plus maximum of BB times over $\mathcal{OS}$ splits.

Observe from Table 3.6 that our standard BB procedure outperformed the triangle splitting method on all but one set of instances, while our $\mathcal{OS}$ splitting procedure outperformed the triangle splitting method on all sets of instances. Also recognize that the total CPU times associated with our $\mathcal{OS}$ splitting procedure are always comparable with those of our standard procedure. We point out that there were many more substantial gaps between solutions to exploit after preprocessing for the instances from [11] than for the instances from [16]. This is the reason that there is a drastic reduction in total number of nodes processed when using $\mathcal{OS}$ splitting on the instances from [11] but not the instances from [16]. We also note that the reported approximate CPU times for a parallel implementation of the $\mathcal{OS}$ splitting procedure indicate that even better results can be obtained once we are able to develop a parallel implementation.

3.5.7 MIPlib Instances

Due to the successful results we obtained using our BB procedure on instances from the literature, we designed our final set of tests to measure the performance of our procedure on a more realistic set of instances. For this we utilized a set of 39 single objective MIP instances available from the MIPlib 2010 library [54]. We chose only instances that were marked easy, were mixed-integer and not pure integer, and were relatively small in size (up to approximately 200 integer variables). For each instance, we generated six secondary objective functions according to the following rules:

(o) For each $i \in \{1, \ldots, m+n\}$ the coefficient c_i^2 is randomly generated from the closed interval $\left[-|c_i^1|, |c_i^1|\right]$.

(a) We solved the LP relaxation associated with f_1 to obtain optimal solution x^*. Then for each nonbasic variable at this solution, we set $c_i^2 = -c_i^1$ if: (i) $c_i^1 > 0$ and x_i^* was not at its lower bound, or (ii) $c_i^1 < 0$ and x_i^* was not at its upper bound. Otherwise we set $c_i^2 = c_i^1$.

(b) We set $c_i^2 = \dfrac{1}{c_i^1}$.

(c) Objective 2 is the sum of the continuous variables.

(d) Objective 2 is the sum of the integer variables, plus one continuous variable.

(e) We solved the LP relaxation associated with f_1 as well as the corresponding MIP. We then repeated strategy (a) for integer variables having the same value at the LP solution as at the MIP solution.

After generation of these instances we did some preliminary testing and discarded instances for which: (i) the Pareto set was a singleton, or (ii) the second objective was unbounded, or (iii) the MIP associated with either f_1 or f_2 took over 12 hours to solve. We then opted to test the performance of the various preprocessing procedures we tested in Section 3.5.2, each set to a maximum execution time of either 5, 10 or 30 minutes. We then calculated the duality gap percentages after exiting preprocessing. The results of this test are displayed in the performance profile found in Figure 3.9. Here "ev" represents the implementation of PreprocessingMethod1 in which ρ is calculated as a function of instance size, "w0" indicates PreprocessingMethod2 with ρ set to zero, and "hy0" indicates the hybrid preprocessing procedure with ρ set to zero. The numbers following each of these represent the limits on execution time.

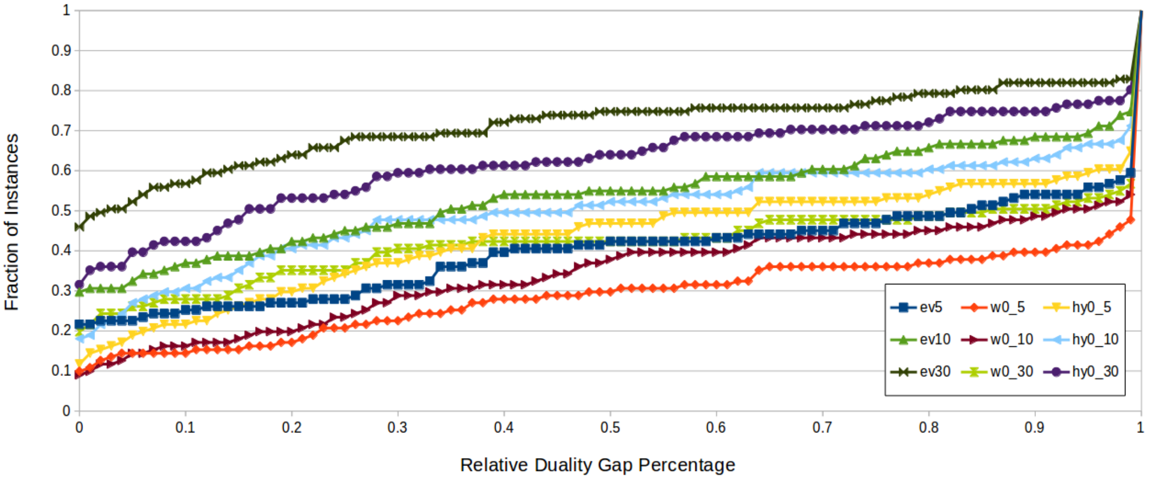


Figure 3.9: Performance profile of duality gap percentage for preprocessing procedures on instances from MIPlib.

Observe from Figure 3.9 that PreprocessingMethod1 with an execution time of 30 minutes performed the best. Hence, we utilized this procedure for our final test. In this test we used our original BB procedure, the BB procedure in which we exploit $\mathcal{OS}$ gaps, and the triangle splitting method on each of the instances generated from MIPlib. We set a maximum time limit of 12 hours for each procedure. Additionally, we set a time limit of 60 seconds for all single objective MIPs solved during the course of our BB procedures, except for: (i) solving the initial MIPs associated with f_1 and f_2, and (ii) solving the MIPs necessary to appropriately reduce subregions of $\mathcal{OS}$ when employing our $\mathcal{OS}$ splitting procedure. We also note that we do not keep track of a duality gap measurement during the course of BB because doing causes a significant reduction in performance, especially when there are a large number of open nodes. Instead, we calculate this gap after termination of BB, but we only allow an additional 12 hours for this task. The results of this experiment are provided in detail in Table 3.7 and summarized by the performance profile in Figure 3.10. Note that when constructing this profile we only included data for instances which were solved by at least one of the three methods for solving BOMILP. Hence, the maximum height of each curve is bounded by the fraction of instances solved in under twelve hours by at least one of the solution procedures.

Table 3.7: Experiment 8 – Comparing with the triangle splitting method for instance from MIPlib.

		Original BB			BB with Gap Splitting				T.S.
Instance		Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	†	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time
aflow40b	o	–	901	*	2,181	2,181	2	–	28,552
	a	–	1,577	*	–	–	1,658	–	–
	d	2,230	1	–	12,099	12,095	166	–	7,234
	e	–	1,943	*	–	–	1,259	*	–
beasleyC3	o	–	–	–	751	749	3	–	–
	a	–	888	14	–	–	9,076	18	–
	b	–	697	24	–	–	533	100	–
	e	–	6,466	*	–	–	12,250	13	–
berlin_5_8_0	1	0	0	0	–	–	–	–	–
bienst2	o	1,596	35	–	1,012	1,011	8	–	2,174
binkar10_1	o	–	4,586	*	–	–	5,461	*	–
	a	3,229	921	–	3,129	3,129	938	–	–
	b	7,634	4,812	–	7,555	7,555	4,433	–	5,949
	c	5,527	2,065	–	8,587	3,238	8,587	–	–
	d	6,892	1,313	–	363	362	2	–	2,360
	e	4,684	286	–	5,393	2,867	522	–	25,293
csched007	o	–	564	31	–	–	596	31	–
	c	–	2,347	2	–	–	2,631	2	–
csched010	o	–	488	35	–	–	395	33	–
	a	–	515	16	–	–	683	16	–
	c	–	1,044	2	15,205	15,205	1	–	–
	d	–	589	100	4,803	4,803	9	–	32,811
danoint	o	42,299	538	–	7,091	7,086	6	–	41,880
dfn-gwin-UUM	o	–	859	67	–	–	1,953	100	25,434
gmu-35-40	o	–	1,527	68	–	–	5,759	69	25,714
	a	2,072	1	–	–	–	2,220	44	–
	d	–	559	2	–	–	719	4	9,973
	e	2,148	1	–	–	–	2,715	44	–
gmu-35-50	o	2,804	1	–	10,218	7,715	1,489	–	–
	a	–	1,234	36	–	–	1,085	49	–
	e	–	907	41	–	–	–	–	–
ic97_potential	o	–	391	71	–	–	446	72	–
	a	–	–	–	–	–	17,248	1	23,911
	b	–	547	70	–	–	1,386	99	–
	d	–	297	97	–	–	340	93	–
k16x240	o	–	17,648	2	–	–	14,740	2	–
	a	–	42,960	6	–	–	23,837	*	–
	b	–	1,022	32	–	–	187,846	86	–
	c	–	2,418	*	–	–	2,479	*	–
	d	–	2,264	*	–	–	3,542	*	–
	e	–	17,251	8	–	–	19,401	–	–
markshare_5_0	o	–	3,905	20	–	–	4,173	13	32,476
	c	–	117,984	2	–	–	166,057	2	14,270
	d	–	3,937	15	5,002	5,002	5	–	–

† – Approximated parallel time
* – duality gap calculation exceeded 12 hours

Table 3.8: (Continuation of Table 3.7.)

		Original BB			BB with Gap Splitting				T.S.
Instance		Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	†	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time
mc11	o	–	620	77	–	–	539	*	–
	a	–	950	*	–	–	6,071	13	–
	c	–	760	36	–	–	275	*	–
	e	–	2,041	*	–	–	918	*	–
mcsched	a	–	14,578	*	–	–	29,378	20	–
	c	–	14,040	0	–	–	19,954	3	–
	d	–	–	–	–	–	9,588	0	–
mik-250-1-100-1	o	–	9,882	6	404	404	3	–	–
	a	–	17,814	12	–	–	21,299	64	–
	c	–	14,170	0	–	–	14,217	*	32,204
	d	–	11,972	*	–	–	8,889	*	–
	e	–	1,434	21	–	–	1,116	59	–
neos-1112787	a	1,931	0	–	1,891	1,890	3	–	–
	c	–	1,881	*	–	–	1,681	*	–
neos-1171737	o	2,112	1	–	2,210	2,210	1	–	–
	d	–	677	33	–	–	673	81	–
neos-1225589	b	–	410	*	–	–	486	*	–
	c	–	1,260	*	–	–	1,291	*	–
	d	–	16,214	*	–	–	44,389	–	–
neos13	o	–	1,779	16	–	–	3,466	100	12,383
	a	–	488	*	–	–	464	*	25,263
	c	496	10	–	574	559	11	–	1,054
	d	–	12,625	36	–	–	170	28	–
neos-1396125	a	824	50	–	825	825	534	–	535
	b	5,648	53	–	3,713	3,590	28	–	3,480
	d	–	613	*	2,111	2,103	4	–	22,695
neos-1426635	d	–	2,761	30	1,203	1,203	2	–	15,155
neos-1426662	d	3,689	1	–	–	–	651	27	–
neos-1440460	d	–	999	35	1,903	1,903	1	–	–
neos-1442657	d	–	576	15	13,097	13,097	1	–	–
neos15	a	–	15,811	34	–	–	18,583	34	–
	b	–	2,911	*	–	–	2,911	42	–
neos-693347	o	–	221	*	–	–	2,670	56	11,480
	c	674	1	–	673	673	1	–	566
	d	722	1	–	718	718	1	–	566
neos-916792	o	–	757	51	–	–	794	49	3,662
	a	985	1	–	–	–	5,257	0	426
	c	–	3,874	2	–	–	3,465	11	6,657
	d	–	672	70	–	–	666	79	–

† – Approximated parallel time

* – duality gap calculation exceeded 12 hours

Table 3.9: (Second continuation of Table 3.7.)

		Original BB			BB with Gap Splitting				T.S.
Instance		Time	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time	†	Nodes	$\overline{\mathcal{G}_{s^*}}$	Time
neos-942830	o	–	1,655	31	–	–	5,606	64	–
	a	8,690	321	–	9,952	9,952	271	–	5,858
	c	3,711	1	–	3,980	3,980	1	–	5,344
	d	4,359	3	–	3,057	3,057	3	–	–
noswot	o	7,491	30,903	–	736	732	3,141	–	1,069
	a	–	–	–	754	753	10	–	–
	c	32,355	5,699	–	–	–	254,933	54	42,748
	d	1,942	1,092	–	713	712	6	–	878
ns1830653	o	23,349	665	–	19,053	19,053	458	–	25,302
	a	19,242	210	–	18,110	18,110	195	–	10,363
p80x400b	o	–	7,024	6	–	–	7,778	6	–
	a	–	16,439	15	–	–	21,374	17	–
	b	–	641	58	–	–	62,581	70	–
	c	–	881	*	–	–	2,205	0	–
	d	–	1,589	*	–	–	2,505	9	–
	e	–	6,803	17	–	–	4,306	24	–
pigeon-10	o	1,940	562	–	1,656	1,656	1	–	1,670
pigeon-11	o	20,436	30	–	20,266	20,266	11	–	20,672
qiu	o	1,665	326	–	1,445	1,445	280	–	4,100
	a	2,726	2,367	–	1,911	873	686	–	1,274
	b	19,706	2,969	–	1,888	1,886	5	–	5,853
	c	–	956	43	1,885	1,864	5	–	4,146
	d	–	41,745	3	1,815	1,815	3	–	–
	e	4,809	541	–	5,642	5,523	4,284	–	834
ran14x18	o	–	2,610	8	–	–	4,424	8	–
	a	–	40,881	11	–	–	56,981	12	–
	b	–	875	33	3,074	3,073	4	–	–
	e	–	46,951	14	–	–	40,132	12	38,624
ran14x18-disj-8	o	–	6,826	10	–	–	9,769	8	–
	a	–	45,528	27	–	–	68,137	28	26,029
	b	–	922	30	–	–	648	1	–
	e	–	10,727	9	–	–	15,977	11	–
ran16x16	o	–	12,180	6	–	–	12,175	6	–
	a	–	23,189	*	–	–	25,738	*	–
	e	–	19,916	11	–	–	9,477	*	–
timtab1	o	–	1,136	50	–	–	1,167	100	–
	a	–	513	100	–	–	535	100	–
	b	–	575	100	–	–	477	100	–
	d	–	789	63	–	–	798	48	–

† – Approximated parallel time

* – duality gap calculation exceeded 12 hours

There are a couple of important pieces of information to recognize from Tables 3.7–3.9. First, notice that of the 115 instance considered, 34 were solved in under 12 hours by the original BB implementation, 43 by the $\mathcal{OS}$ splitting BB variant, and 40 by the traingle splitting method. Additionally, there were 17 instances which were solved in under 12 hours by one version of BB, but not by the triangle splitting method, and 10 instances solved in under 12 hours by the triangle splitting method, but not by a BB procedure. In all, the results display comparable performance between the BB approaches and the triangle splitting method. This pattern is also supported by the profile in Figure 3.10. We also point out that there are a small number of instances for which one of the BB procedures terminated after processing a very small number of nodes. There are two situations in which this occurred: (i) when all Pareto solutions on a BOMILP instance lie on a single line segment in $\mathcal{OS}$, and (ii) when there are an extremely low number of Pareto points or line segments. The former case seems to happen far less frequently than the latter, but it should be noted that in this case numerical issues can cause BB to terminate before all Pareto solutions are found if a cutting plane is generated which lies on the same segment in $\mathcal{OS}$ on which all Pareto solutions lie.

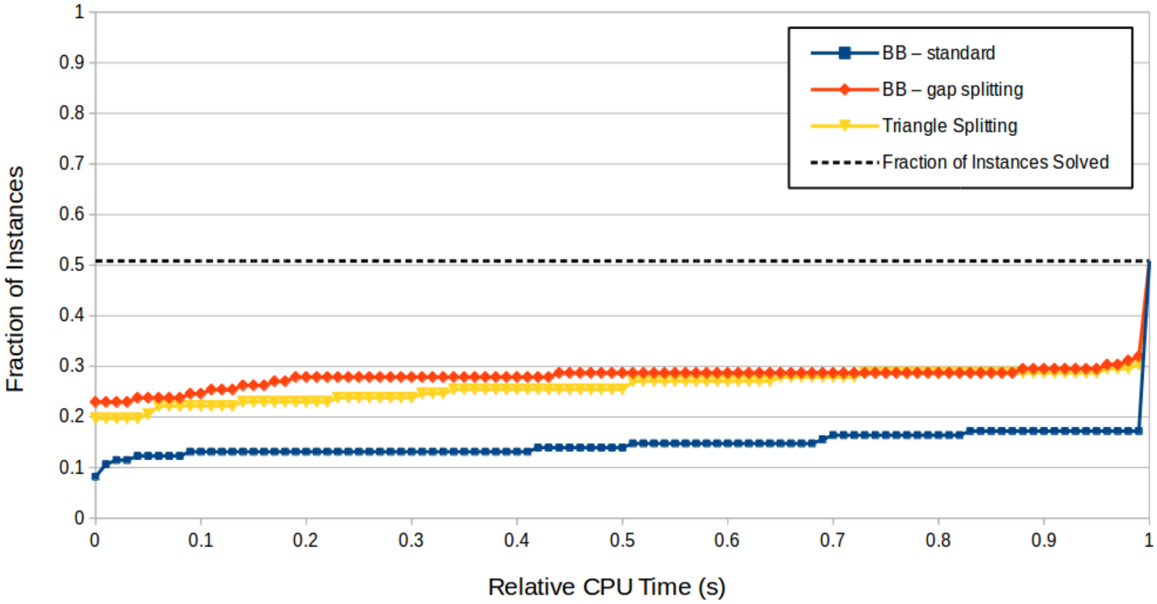


Figure 3.10: Performance profile of relative CPU time for MIPlib instances.

3.6 Conclusion

In this work we have introduced a new BB scheme for solving BOMILP with general integers. We have studied each of the key components of single objective BB and presented a strategy for extending each component to a biobjective framework. We have also conducted a number of computational experiments. The first several experiments provide insight into the usefulness of each of the tools we have presented. The final few experiments compare the performance of our BB procedure and the triangle splitting method [16]. Our BB procedure outperforms the triangle splitting method on small instances which were previously considered in the literature, and performs comparably to the triangle splitting method on a new set of large, challenging instances which we developed using instances of single objective MIP from MIPlib 2010 [54].

Chapter 4

A Two-phase Algorithm for mpLCP with Parameters in the q Vector

[The contents of this chapter include material from a paper entitled "A two-phase algorithm for the multiparametric linear complementarity problem," which was submitted to the *European Journal on Operational Research* in January of 2015; the authors are N. Adelgren and M. M. Wiecek. The paper was accepted for publication in April of 2016.]

4.1 Introduction

We consider a parametric form of the Linear Complementarity Problem (LCP) in which the right hand side vector is dependent on a vector of parameters $\theta \in S_\theta \subseteq \mathbb{R}^k$, where S_θ is a bounded convex polytope defining the set of "attainable" values for θ. This problem, referred to as the multiparametric Linear Complementarity Problem (mpLCP), is as follows:

> Given $M \in \mathbb{R}^{h\times h}$, $q \in \mathbb{R}^h$ and $\triangle Q \in \mathbb{R}^{h\times k}$ $(k \leq h)$, for each $\theta \in S_\theta$ find vectors $w(\theta)$ and $z(\theta)$ in $\mathbb{R}^h$ that satisfy the following system:

$$\begin{aligned} w - Mz &= q + \triangle Q\theta \\ w^\top z &= 0 \\ w, z &\geq 0 \end{aligned} \tag{4.1}$$

If such a solution exists for a given $\theta \in S_\theta$, mpLCP is said to be *feasible at* θ, otherwise it is *infeasible at* θ. Similarly, mpLCP is said to be *feasible* if there exists a $\hat{\theta} \in S_\theta$ at which mpLCP is feasible, otherwise mpLCP is *infeasible.* As finding a solution to (4.1) for each $\theta \in S_\theta$ individually is intractable, the goal of mpLCP is to partition the space S_θ into regions such that the representation of the solution vectors w and z as functions of θ is invariant over each region. In the literature these regions have been given a variety of names, such as invariancy regions, critical regions, and validity sets. We refer to them as *invariancy regions* and discuss them in more detail in the next section.

LCP, and by extension mpLCP, has numerous applications in the fields of engineering and economics. For an extensive list we suggest [22, 65]. It is well known that Linear Programs (LPs) Quadratic Programs (QPs) with convex objective functions and linear constraints can be reformulated as LCPs. Thus, mpLCP encompasses multiparametric LPs (mpLPs) and multiparametric QPs (mpQPs) containing parameters in the linear term of the objective function and in the right hand sides of the constraints. Recently mpQPs of this form have received much attention in the literature for their application to model predictive control [5, 12, 39, 40, 72, 74, 87, 88, 94, 95].

Another important class of problems that has received considerable attention in recent years and can also be formulated as a mpQP is multiobjective optimization problems with a single psuedoconvex objective and any number of linear objectives. These types of problems are particularly relevant in the areas of economics and finance. Examples of works considering these types of problems include [44, 76, 84, 104, 105] and the references therein.

In general LCP is NP-hard, though polynomial time algorithms exist for certain classes of the matrix M. Thus, much work has been done in order to identify various classes of matrices M which impact one's ability to solve an instance of LCP. Solution techniques for LCP are often designed for specific classes of M. For a concise list of important matrix classes see [21]. For a detailed discussion on these classes and their impact on LCP see [22, 65]. We will refer to many of the matrix classes discussed in these works throughout this chapter. As the method we proposed requires that M be a sufficient matrix, we provide the following definition, as found in [22].

Definition 4.1. A matrix $M \in \mathbb{R}^{h\times h}$ is *column sufficient* if the following implication is satisfied:

$$\{x_i(Mx)_i \leq 0 \text{ for all } i\} \quad \Rightarrow \quad \{x_i(Mx)_i = 0 \text{ for all } i\} \tag{4.2}$$

M is said to be *row sufficient* if $M^\top$ is column sufficient. If M is both column and row sufficient, it is then called *sufficient.*

Parametric LCP with a single parameter (i.e., $k = 1$) has been studied quite extensively. Some of the works considering this problem include Cottle [20], Danao [24], Pang [69] and Pang et al. [70]. Columbano et al. [18], Gailly et al. [32], Jones and Morari [46] and Li and Ierapetritou [57] consider mpLCP as in (4.1) (i.e., $k > 1$). The method of Gailly et al. [32] is designed for the case in which M is copositive-plus. The method is theoretically sound but lacks a practical discussion as to how the theory should be implemented. Jones and Morari [46] propose a method for the case in which M is positive semi-definite. Their method is an extension of techniques that are used for solving single parametric LCP, but depends on a lexicographic ϵ-perturbation in order to handle degeneracy. Columbano et al. [18] developed a technique for instances in which M is a sufficient matrix. When certain conditions are not satisfied, however, their method depends on an ϵ-perturbation technique in which an auxiliary multiobjective program must be solved. The method of Li and Ierapetritou [57] works for general M, but is computationally expensive since it requires reformulating the mpLCP as a multiparametric bilinear mixed integer program. Recently, Herceg et al. [42] proposed a technique designed for general M which extends the enumerative approach of [40] for solving mpQP to the context of mpLCP.

Significant improvements can still be made on solution techniques for mpLCP. In this chapter we propose a two-phase technique for solving instances of mpLCP in which M is sufficient. Phase 1 is used for initialization and only terminates when: (i) an instance of mpLCP has been shown to be infeasible, or (ii) an initial feasible solution and the corresponding invariancy region have been discovered. In the latter case, Phase 2 is then used to partition S_θ. Phase 2 is inspired by the work of Columbano et al. [18], but does not rely on an ϵ-perturbation technique and therefore has an improved worst-case complexity. We point out that in our consideration of Phase 1 we answer a very important question that no other work we are aware of has considered, the question of how one can determine an initial feasible solution for a (multi)parametric LCP problem. In all works we know of, it is simply assumed that such a solution is available.

As mentioned, the method for solving mpLCP which we present in this work is a two-phase method. We will show that the problem solved in the first phase of this method is a special case of the problem solved during the second phase. For this reason we discuss Phase 2 prior to Phase 1. Hence, the remainder of this chapter is organized as follows. Background information on LCP problems and their geometrical structure is contained in Section 4.2. The theory and methodology for Phase 2 of the proposed method for solving mpLCP are presented in Section 4.3. In Section 4.3

we present the theory and methodology for Phase 1. We discuss the complexity of each algorithm and present numerical results for applying the proposed two-phase method to a collection of mpQP instances in Section 4.5. In Section 4.6 we provide concluding remarks and a discussion on proposed future work. In Section 4.A we offer an illustrative example, showing explicitly how the Phase 1 and 2 algorithms are used to solve an instance of mpLCP. Section 4.B contains detailed results from our computational experiments as well as a couple of supporting images.

4.2 Background on mpLCP

This section is divided into two subsections. In the first we present preliminary notations and definitions and in the second we provide a discussion on the geometry of mpLCP and provide some preliminary results.

4.2.1 Preliminaries

We begin this subsection by introducing definitions and notation necessary for the remainder of this work. Assume that we are given an mpLCP of the form (4.1) and define the matrix $G := \begin{bmatrix} I & -M \end{bmatrix}$ and the vector $\nu := \begin{bmatrix} w \\ z \end{bmatrix}$, where $G \in \mathbb{R}^{h \times 2h}$ and $\nu \in \mathbb{R}^{2h}$. We use the notation $G_{i\bullet}$ to represent the i^{th} row of G and $G_{\bullet j}$ to represent the j^{th} column of G. Also, given a set $\mathcal{I} \subseteq \{1, \ldots, h\}$ we use $G_{\mathcal{I}\bullet}$ to denote the matrix formed by the rows of G indexed by $\mathcal{I}$. Similarly, given a set $\mathcal{J} \subseteq \{1, \ldots, 2h\}$ we use $G_{\bullet\mathcal{J}}$ to denote the matrix formed by the columns of G indexed by $\mathcal{J}$. Furthermore, given $\mathcal{I} \subseteq \{1, \ldots, h\}$ and $\mathcal{J} \subseteq \{1, \ldots, 2h\}$, we use $G_{\mathcal{I}\mathcal{J}}$ to represent the submatrix of G consisting of the elements of the rows indexed by $\mathcal{I}$ which are in the columns indexed by $\mathcal{J}$, i.e., $G_{\mathcal{I}\mathcal{J}} = (G_{\mathcal{I}\bullet})_{\bullet\mathcal{J}}$. Let $\mathcal{E}$ denote the index set $\{1, \ldots, 2h\}$ for (4.1).

Definition 4.2. A *basis* is a set $\mathcal{B} \subset \mathcal{E}$ such that $|\mathcal{B}| = h$ and $rank(G_{\bullet\mathcal{B}}) = h$. The set $\mathcal{N} := \mathcal{E} \setminus \mathcal{B}$ is called the *complement* of $\mathcal{B}$.

Definition 4.3. The sets of variables $\nu_{\mathcal{B}} := \{\nu_i : i \in \mathcal{B}\}$ and $\nu_{\mathcal{N}} := \{\nu_i : i \in \mathcal{N}\}$ are referred to as the sets of *basic* and *nonbasic* variables, respectively.

Definition 4.4. Given a basis $\mathcal{B}$, for every $\theta \in S_\theta$, $\nu_{\mathcal{B}}(\theta) = G_{\bullet\mathcal{B}}^{-1}\left(q + \triangle Q\theta\right)$, $\nu_{\mathcal{N}}(\theta) = 0$ is a solution to the linear system: $G\nu = q + \triangle Q\theta$. For each $\theta \in S_\theta$, the solution $(\nu_{\mathcal{B}}(\theta), \nu_{\mathcal{N}}(\theta))$ is called a *basic solution.*

Definition 4.5. A basis $\mathcal{B}$ is called *complementary* if $|\{i, i+h\} \cap \mathcal{B}| = 1$ for each $i \in \{1, \ldots, h\}$.

We have now built the tools necessary for providing the definition of an *invariancy region*. Consider a complementary basis $\mathcal{B}$ and suppose there exists $\theta \in S_\theta$ such that: (i) $\nu_{\mathcal{B}}(\theta) = G^{-1}_{\bullet\mathcal{B}}(q + \triangle Q\theta) \geq 0$ and, (ii) $\nu_{\mathcal{N}}(\theta) = 0$. Then since $\nu = \begin{bmatrix} w \\ z \end{bmatrix}$, for all $\theta \in S_\theta$ satisfying (i) and (ii) above, the basic solution $(\nu_{\mathcal{B}}(\theta), \nu_{\mathcal{N}}(\theta))$ satisfies (4.1) and therefore defines solution vectors $w(\theta)$ and $z(\theta)$ for mpLCP. Note that one set of solution vectors of this form may exist for each complementary basis.

Definition 4.6. The *invariancy region* $\mathcal{IR}_{\mathcal{B}}$ of a complementary basis $\mathcal{B}$ is the set:

$$\mathcal{IR}_{\mathcal{B}} := \left\{\theta \in S_\theta : G^{-1}_{\bullet\mathcal{B}}(q + \triangle Q\theta) \geq 0\right\} \tag{4.3}$$

Hence, there may exist one invariancy region for each complementary basis.

Definition 4.7. A complementary basis $\mathcal{B}$ is called *feasible* to (4.1) if $\mathcal{IR}_{\mathcal{B}} \neq \emptyset$.

Every invariancy region is a convex polytope contained within S_θ. For every feasible complementary basis $\mathcal{B}$, the affine function defined by $\nu_{\mathcal{B}}(\theta) = G^{-1}_{\bullet\mathcal{B}}(q + \triangle Q\theta)$, $\nu_{\mathcal{N}}(\theta) = 0$ is a solution to (4.1) for all $\theta \in \mathcal{IR}_{\mathcal{B}}$. Therefore in this work we propose a method for determining a piecewise affine solution to (4.1) by partitioning S_θ into a set of invariancy regions with disjoint relative interiors.

Definition 4.8. For an arbitrary set S, the relative interior of S is the set $relint(S) := \{s \in S : \exists \epsilon > 0, B_\epsilon(s) \cap aff(S) \subseteq S\}$, where $B_\epsilon(s)$ is the ball of radius ϵ centered at s and $aff(S)$ is the affine hull of S, i.e., the intersection of all affine sets containing S.

Definition 4.9. Given a feasible complementary basis $\mathcal{B}$, the invariancy region $\mathcal{IR}_{\mathcal{B}}$ is called *full-dimensional* if $dim\left(\mathcal{IR}_{\mathcal{B}}\right) = k$.

Definition 4.10. Given distinct feasible complementary bases $\mathcal{B}$ and $\mathcal{B}'$, the invariancy regions $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are called *adjacent* if $dim\left(\mathcal{IR}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}\right) = k - 1$.

4.2.2 Geometry of the mpLCP

We divide this subsection in two. First we discuss some properties of the mpLCP problem that we will need in order to establish an algorithm for partitioning S_θ. After this discussion we provide theoretical justification for some of the assumptions made.

4.2.2.1 The relationship between invariancy regions and complementary cones

We now give several definitions and properties needed for this discussion.

Definition 4.11. For an index $i \in \mathcal{E}$ the *complementary index* of i is $\bar{\imath} := (i + h) \mod 2h$.

Similarly, given a set $\mathcal{I} \subseteq \mathcal{E}$, the set $\overline{\mathcal{I}}$ is defined as the set of all complementary indices of elements in $\mathcal{I}$.

Definition 4.12. A set $\mathcal{J} \subset \mathcal{E}$ is called *complementary* if $|\{i, i+h\} \cap \mathcal{J}| = 1 \; \forall i \in \{1, \ldots, h\}$, i.e., if $i \in \mathcal{J} \Rightarrow \bar{\imath} \notin \mathcal{J}$.

Given an arbitrary matrix P, we use the notation *span*(P) to represent the set of all linear combinations of the columns of P. Similarly, we use the notation *cone*(P) to represent the set of all nonnegative combinations of the columns of P. Hence, the dimension of *cone*(P), denoted *dim*(*cone*(P)), is the number of linearly independent columns of P.

Definition 4.13. For any complementary set $\mathcal{J}$, the set $\mathcal{C}(\mathcal{J}) := cone\,(G_{\bullet\mathcal{J}})$ is called a *complementary cone* with respect to the matrix M. The union of all such cones is called the *complementary range* of M and is denoted $\mathcal{K}(M)$.

Definition 4.14. A complementary cone $\mathcal{C}(\mathcal{J})$ is called *full-dimensional* if $dim\,(\mathcal{C}(\mathcal{J})) = h$, i.e., if $rank\,(G_{\bullet\mathcal{J}}) = h$.

Proposition 4.15. A complementary cone $\mathcal{C}(\mathcal{J})$ is full-dimensional if and only if $\mathcal{J}$ is a complementary basis.

Proof. ($\Rightarrow$): Since $\mathcal{C}(\mathcal{J})$ is a complementary cone, $\mathcal{J}$ is complementary and thus $|\mathcal{J}| = h$. Since $\mathcal{C}(\mathcal{J})$ is full-dimensional, $rank\,(G_{\bullet\mathcal{J}}) = h$.
($\Leftarrow$): Since $\mathcal{J}$ is a complementary basis, $rank\,(G_{\bullet\mathcal{J}}) = h$ and so $dim\,(\mathcal{C}(\mathcal{J})) = h$. □

Definition 4.16. For distinct complementary bases $\mathcal{B}_1$ and $\mathcal{B}_2$, the complementary cones $\mathcal{C}(\mathcal{B}_1)$ and $\mathcal{C}(\mathcal{B}_2)$ are called *adjacent* if $dim\,(\mathcal{C}(\mathcal{B}_1) \cap \mathcal{C}(\mathcal{B}_2)) = h - 1$. In this case the bases $\mathcal{B}_1$ and $\mathcal{B}_2$ are also called adjacent.

For any complementary basis $\mathcal{B}$, the associated complementary cone is:

$$\mathcal{C}(\mathcal{B}) = \left\{\tau \in \mathbb{R}^h : G_{\bullet\mathcal{B}}^{-1}\tau \geq 0\right\} \tag{4.4}$$

Thus $\mathcal{C}(\mathscr{B})$ defines the set of all vectors $q + \triangle Q\theta$ for which the basis $\mathscr{B}$ is feasible and $\mathcal{K}(M)$ defines the set of all vectors $q + \triangle Q\theta$ for which any basis is feasible. Note the close relationship between $\mathcal{IR}_{\mathscr{B}}$ and $\mathcal{C}(\mathscr{B})$. In order to understand this relationship further, we now consider the specifics of the affine subspace:

$$\mathscr{AS} := \{q + \triangle Q\theta : \theta \in S_\theta\} \tag{4.5}$$

When considering $\mathscr{AS}$ we make the assumption that $\triangle Q$ is of full column rank. In their work, Columbano et al. [18] also make this assumption, but do not provide any justification. We proceed with this discussion and utilize this assumption, but theoretical justification is provided in Section 4.2.2.2.

Definition 4.17. Given a complementary basis $\mathscr{B}$, the *invariant domain of* $\mathscr{B}$, denoted $\mathcal{ID}_{\mathscr{B}}$, is the intersection of the affine subspace $\mathscr{AS}$ with the complementary cone $\mathcal{C}(\mathscr{B})$. Thus $\mathcal{ID}_{\mathscr{B}} := \mathcal{C}(\mathscr{B}) \cap \mathscr{AS} = \left\{\tau : G^{-1}_{\bullet\mathscr{B}}\tau \geq 0, \tau = q + \triangle Q\theta, \theta \in S_\theta\right\}$.

Since we have assumed that $\triangle Q$ is of full column rank, we can view the function $\theta \to q + \triangle Q\theta$ as a bijective function. As a result, we make the following two observations:

Observation 4.18. For a given basis $\mathscr{B}$, we have $\mathcal{ID}_{\mathscr{B}} = q + \triangle Q\,(\mathcal{IR}_{\mathscr{B}})$.

Observation 4.19. For each $i \in \mathscr{E}$, the inequality $\left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet}\tau \geq 0$ is redundant in $\mathcal{ID}_{\mathscr{B}}$ if and only if $\left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet}(q + \triangle Q) \geq 0$ is redundant in $\mathcal{IR}_{\mathscr{B}}$.

Notice that these observations show explicitly the relationship existing between invariancy regions and complementary cones.

It is well known that given an instance of LCP, and by extension mpLCP, the properties of the matrix M have a significant impact on the structure of the complementary cones, and hence also the invariancy regions in the case of mpLCP. Of particular importance for this discussion are column sufficient matrices, which we defined previously, and Q_0 matrices, for which we quote the definition from [65].

Definition 4.20. The square matrix M is said to be a Q_0 matrix if $\mathcal{K}(M)$ is a convex cone.

Recognize from Definition 4.20 that since each complementary cone is convex, if $\mathcal{K}(M)$ is convex then $\mathscr{AS} \cap \mathcal{K}(M)$ is also convex. Thus, every nonempty invariant domain has an adjacent nonempty invariant domain. We now recall the following important property of column sufficient matrices, shown in [22]: if M is column sufficient, all complementary cones have disjoint relative interiors. The benefit of this property is that if the relative interiors of all complementary cones are disjoint, then S_θ can be partitioned since no two invariant domains can have an intersection of dimension larger than $h-1$. Thus, the algorithms we present in Section 4.5 are designed for instances of mpLCP in which M which is both Q_0 and column sufficient. Note that the largest class of matrices known to be a subset of both the classes of Q_0 and column sufficient matrices is the class of sufficient matrices.

4.2.2.2 Theoretical justification for the assumption that $\triangle Q$ is of full column rank

We point out to the reader that although the following discussion is theoretically important, it is not necessary for understanding the algorithms presented in Section 4.5 and can therefore be passed over if so desired.

We will show that in the case when $\triangle Q$ is not full column rank, there exists a "reduced" mpLCP, which we denote mpLCP$'$, which can be solved in place of the original mpLCP. Furthermore, we show that the solution of mpLCP can be fully recovered from the solution of mpLCP$'$.

Consider the matrix $\triangle Q \in \mathbb{R}^{h\times k}$ and suppose $rank(\triangle Q) = \ell < k$. Let $L \subset \{1,\dots,h\}$ be an index set such that $|L| = \ell$ and the columns of $(\triangle Q)_{\bullet L}$ are linearly independent. We obtain mpLCP$'$ from (4.1) by substituting $(\triangle Q)_{\bullet L}\,\varphi$ for $\triangle Q\theta$:

$$\begin{gathered} w - Mz = q + (\triangle Q)_{\bullet L}\,\varphi \\ w^\top z = 0 \\ w, z \geq 0 \end{gathered} \tag{4.6}$$

Here $\varphi \in S_\varphi \subset \mathbb{R}^\ell$ is a new set of parameters. We now discuss the relationship between solutions to (4.1) and (4.6) and how a solution for one can be obtained from a solution to the other. First consider the set of linear equations

$$(\triangle Q)_{\bullet L}\,\varphi = \triangle Q\theta \tag{4.7}$$

Here φ and θ can be treated as variables. Each can be fixed and the other solved for. Since $(\triangle Q)_{\bullet L}$ is of full column rank, it has a left inverse, so from (4.7), we obtain

$$\varphi = \left((\triangle Q)_{\bullet L}^\top (\triangle Q)_{\bullet L}\right)^{-1} (\triangle Q)_{\bullet L}^\top \triangle Q\theta. \tag{4.8}$$

Thus, every $\theta \in S_\theta$ maps to a $\varphi \in \mathbb{R}^\ell$. We define this mapping explicitly as:

$$\mathcal{T}(\cdot) : \mathbb{R}^k \to \mathbb{R}^\ell \quad \text{such that} \quad \mathcal{T}(\theta) := \left((\triangle Q)_{\bullet L}^\top (\triangle Q)_{\bullet L} \right)^{-1} (\triangle Q)_{\bullet L}^\top \triangle Q \theta \tag{4.9}$$

Hence, for each $\theta \in S_\theta$ we have $\varphi_\theta := \mathcal{T}(\theta)$, and also $S_\varphi := \mathcal{T}(S_\theta)$. Let $\Lambda = \{1, \ldots, h\} \backslash L$. By rearranging the columns of $\triangle Q$ and the corresponding rows of θ, we have $\triangle Q \theta = \begin{bmatrix} \triangle Q_{\bullet L} & \triangle Q_{\bullet \Lambda} \end{bmatrix} \begin{bmatrix} \theta_{L\bullet} \\ \theta_{\Lambda\bullet} \end{bmatrix}$. By replacing $\triangle Q \theta$ with $\begin{bmatrix} \triangle Q_{\bullet L} & \triangle Q_{\bullet \Lambda} \end{bmatrix} \begin{bmatrix} \theta_{L\bullet} \\ \theta_{\Lambda\bullet} \end{bmatrix}$ in (4.9), we obtain the following alternate form of $\mathcal{T}(\theta)$:

$$\mathcal{T}(\theta) = \theta_{L\bullet} + \tilde{Q} \theta_{\Lambda\bullet} \tag{4.10}$$

where $\tilde{Q} = \left((\triangle Q)_{\bullet L}^\top (\triangle Q)_{\bullet L} \right)^{-1} (\triangle Q)_{\bullet L}^\top \triangle Q_{\bullet \Lambda}$.

Although each $\theta \in S_\theta$ maps to a unique $\varphi \in S_\varphi$, the converse does not hold. For each $\varphi \in S_\varphi$ consider the set:

$$U_\varphi := \left\{ \theta \in \mathbb{R}^k : \varphi \text{ and } \theta \text{ satisfy } (\triangle Q)_{\bullet L} \varphi = \triangle Q \theta \right\} \tag{4.11}$$

Recall that $S_\theta \subset \mathbb{R}^k$. Since $S_\varphi = \mathcal{T}(S_\theta)$, for each $\varphi \in S_\varphi$ there is at least one $\theta \in S_\theta$ satisfying (4.7). Furthermore, since one such θ exists and $rank\,(\triangle Q) = \ell < k$, for each $\varphi \in S_\varphi$ there exist an infinite number of values of θ which satisfy (4.7). Thus, for a given $\hat{\varphi} \in S_\varphi$, a $\hat{\theta} \in U_\varphi$ can be found by selecting arbitrary values for $k - \ell$ elements of $\hat{\theta}$ and solving for the remaining elements using the equations in (4.7). Without loss of generality, we select the elements of $\hat{\theta}$ whose indices are in Λ to be the arbitrarily selected values. Letting $\hat{\theta}_{\Lambda\bullet} = \hat{\chi} \in \mathbb{R}^{k-l}$ and solving (4.7) for the remaining elements of $\hat{\theta}$ gives:

$$\hat{\theta}_{L\bullet} = \hat{\varphi} - \tilde{Q} \hat{\chi} \tag{4.12}$$

It is easy to verify that these values satisfy the system $(\triangle Q)_{\bullet L} \hat{\varphi} = \triangle Q \hat{\theta}$. Thus, every pair $(\chi, \varphi) \in \mathbb{R}^{k-\ell} \times S_\varphi$ maps to a $\theta \in U_\varphi$ by letting $\begin{bmatrix} \theta_{L\bullet} \\ \theta_{\Lambda\bullet} \end{bmatrix} = \begin{bmatrix} \varphi - \tilde{Q}\chi \\ \chi \end{bmatrix}$. This shows that

$$U_\varphi = \left\{ \theta \in \mathbb{R}^k : \theta_{L\bullet} = \varphi - \tilde{Q}\chi \text{ and } \theta_{\Lambda\bullet} = \chi \text{ for any } \chi \in \mathbb{R}^{k-\ell} \right\}. \tag{4.13}$$

Now that we have discussed the relationships between $\theta \in S_\theta$ and $\varphi \in S_\varphi$, we are ready to show that the solution of mpLCP can be fully recovered from the solution of mpLCP′. Notice that for every complementary feasible basis $\mathcal{B}$ and invariancy region $\mathcal{IR}_{\mathcal{B}}$ of mpLCP, there is a corresponding invariancy region

$$\mathcal{IR}'_{\mathcal{B}} := \left\{\varphi \in S_\varphi \subseteq \mathbb{R}^\ell : G^{-1}_{\bullet\mathcal{B}} \left(q + (\triangle Q)_{\bullet L}\, \varphi\right) \geq 0\right\} \tag{4.14}$$

of mpLCP$'$. Thus, a partition of S_θ can be derived from a partition of S_φ if the following hold:

- An invariancy region $\mathcal{IR}_{\mathcal{B}}$ is full dimensional if and only if $\mathcal{IR}'_{\mathcal{B}}$ is also full dimensional.
- Two invariancy regions $\mathcal{IR}_{\mathcal{B}_1}$ and $\mathcal{IR}_{\mathcal{B}_2}$ are adjacent if and only if $\mathcal{IR}'_{\mathcal{B}_1}$ and $\mathcal{IR}'_{\mathcal{B}_2}$ are adjacent.

Consider the following lemmas and the subsequent propositions.

Lemma 4.21. If $\hat{\varphi} = \mathcal{T}(\hat{\theta})$ lies on the boundary of S_φ then $\hat{\theta}$ lies on the boundary of S_θ.

Proof. We prove the contrapositive. Notice that the following hold:
(i) Since $\hat{\theta}$ does not lie on the boundary of S_θ, for all vectors θ^i in S_θ, there exists $\epsilon_i > 0$ such that $\hat{\theta} + \epsilon_i\theta^i \in S_\theta$.
(ii) Since $S_\varphi = \mathcal{T}(S_\theta)$, for all vectors φ^i in S_φ there must exist $\theta^i \in S_\theta$ and $\delta_i > 0$ such that $\varphi^i = \delta_i\mathcal{T}(\theta^i)$.
Now, (i) implies that $\mathcal{T}(\hat{\theta} + \epsilon_i\theta^i) = \hat{\varphi} + \epsilon_i\mathcal{T}(\theta^i) \in S_\varphi$ for all vectors $\theta^i \in S_\theta$. From this result and (ii) we conclude that $\hat{\varphi} + \dfrac{\epsilon_i}{\delta_i}\varphi^i \in S_\varphi$ for all vectors $\varphi^i \in S_\varphi$. Therefore $\hat{\varphi}$ cannot lie on the boundary of S_φ. □

Lemma 4.22. If $\hat{\varphi}$ lies in the relative interior of S_φ then there exists $\hat{\theta}$ in the relative interior of S_θ such that $\hat{\varphi} = \mathcal{T}(\hat{\theta})$.

Proof. We again prove the contrapositive. Assume that there does not exist $\hat{\theta} \in relint(S_\theta)$ such that $\mathcal{T}(\hat{\theta}) = \hat{\varphi}$. We do not consider the case in which there does not exist $\theta \in S_\theta$ such that $\mathcal{T}(\theta) = \hat{\varphi}$ since it is trivial. Thus, assume that every $\theta^i \in S_\theta$ such that $\mathcal{T}(\theta^i) = \hat{\varphi}$ is on the boundary of S_θ. In this case, all θ^i such that $\mathcal{T}(\theta^i) = \hat{\varphi}$ must lie on a single $k-1$ dimensional facet F of S_θ. To see this, suppose there exist $\theta^1, \theta^2 \in S_\theta$ such that $\mathcal{T}(\theta^1) = \mathcal{T}(\theta^2) = \check{\varphi}$ and notice that for all $\lambda \in (0,1)$, $\mathcal{T}(\lambda\theta^1 + (1-\lambda)\theta^2) = \lambda\check{\varphi} + (1-\lambda)\check{\varphi} = \check{\varphi}$. Hence, if θ^1 and θ^2 did not both lie on F, there would exist a $\lambda \in (0,1)$ such that $(\lambda\theta^1 + (1-\lambda)\theta^2) \in relint(S_\theta)$, which contradicts the original assumption.

Now, recognize from (4.11) that for each $\varphi \in S_\varphi$, U_φ is a $k-h$ dimensional hyperplane. Then from our previous arguments we can conclude that $U_{\check{\varphi}} \cap S_\theta \subset F$. Let H^* be the $k-1$ dimensional hyperplane containing F, θ' be a point in F for which $\mathcal{T}(\theta') = \hat{\varphi}$, and $\tilde{\theta}$ be a vector in S_θ which is normal to F and oriented so that it points away from the interior of S_θ. Then, as a direct corollary to the Separating Hyperplane Theorem (Thm 2.4.4, pp. 53 [8]), for all $\epsilon > 0$, H^* separates $U_{\mathcal{T}(\theta'+\epsilon\theta)}$ from S_θ (since for distinct $\varphi^1, \varphi^2 \in S_\varphi$, U_{φ^1} and U_{φ^2} are parallel), i.e., $U_{\mathcal{T}(\theta'+\epsilon\theta)} \cap S_\theta = \emptyset$. This implies that for all $\epsilon > 0$, $\mathcal{T}(\theta' + \epsilon\tilde{\theta}) = \hat{\varphi} + \epsilon\mathcal{T}(\tilde{\theta}) \notin S_\varphi$. Therefore there cannot exist $\epsilon > 0$ such that $B_\epsilon(\hat{\varphi}) \subseteq S_\varphi$. Hence $\hat{\varphi} \notin relint(S_\varphi)$. □

Proposition 4.23. An invariancy region $\mathcal{IR}_{\mathcal{B}}$ is full dimensional if and only if $\mathcal{IR}'_{\mathcal{B}}$ is also full dimensional.

Proof. ($\Leftarrow$): Since $\mathcal{IR}'_{\mathcal{B}} \subset \mathbb{R}^\ell$ is full dimensional it must contain $\ell+1$ affinely independent points, $\varphi^1, \ldots, \varphi^{\ell+1}$. There must also exist vectors $\chi^1, \ldots, \chi^{k-\ell} \in \mathbb{R}^{k-\ell}$ which are linearly independent. We show that $\varphi^1, \ldots, \varphi^{\ell+1}$ and $\chi^1, \ldots, \chi^{k-\ell} \in \mathbb{R}^{k-\ell}$ can be used to construct $k+1$ affinely independent points $\theta^1, \ldots, \theta^{k+1}$ in $\mathcal{IR}_{\mathcal{B}} \subset S_\theta \subseteq \mathbb{R}^k$. We first construct these points and then show the affine independence.

Let $\varphi^* \in relint\left(\mathcal{IR}'_{\mathcal{B}}\right)$. By Lemmas 3.1 and 3.2, there must exist $\chi^* \in \mathbb{R}^{k-\ell}$ such that θ^* defined as $\begin{bmatrix}\theta^*_{L\bullet}\\ \theta^*_{\Lambda\bullet}\end{bmatrix} = \begin{bmatrix}\varphi^* - \tilde{Q}\chi^*\\ \chi^*\end{bmatrix} \in relint(S_\theta)$. We now show the construction of each θ^i by first considering $i \in \{1, \ldots, \ell+1\}$ and then $i \in \{\ell+2, \ldots, k+1\}$.

Case 1: $i \in \{1, \ldots, \ell+1\}$

Since $\varphi^* \in relint\left(\mathcal{IR}'_{\mathcal{B}}\right)$, for each $i \in \{1, \ldots, \ell+1\}$ there exists an $\epsilon_i > 0$ such that $\left(\varphi^* + \delta_i\varphi^i\right) \in \mathcal{IR}'_{\mathcal{B}}$ for all $\delta_i \in [0, \epsilon_i]$. Also, since $\theta^* \in relint(S_\theta)$, for each $i \in \{1, \ldots, \ell+1\}$ there must exist an $\hat{\epsilon}_i > 0$ such that $\begin{bmatrix}\theta^*_{L\bullet}\\ \theta^*_{\Lambda\bullet}\end{bmatrix} + \hat{\delta}_i\begin{bmatrix}\varphi^i\\ 0\end{bmatrix} \in S_\theta$ for all $\hat{\delta}_i \in [0, \hat{\epsilon}_i]$. For each $i \in \{1, \ldots, \ell+1\}$, define θ^i so that

$$\begin{bmatrix}\theta^i_{L\bullet}\\ \theta^i_{\Lambda\bullet}\end{bmatrix} = \begin{bmatrix}\varphi^* + \epsilon^*_i\varphi^i - \tilde{Q}\chi^*\\ \chi^*\end{bmatrix} \tag{4.15}$$

where $\epsilon^*_i := \min\{\epsilon_i, \hat{\epsilon}_i\}$. Then for all $i \in \{1, \ldots, \ell+1\}$, $\left(\varphi^* + \epsilon^*_i\varphi^i\right) \in \mathcal{IR}'_{\mathcal{B}}$ and $\begin{bmatrix}\theta^i_{L\bullet}\\ \theta^i_{\Lambda\bullet}\end{bmatrix} \in S_\theta$.

We now show that for each $i \in \{1, \ldots, \ell+1\}$, $\theta^i \in \mathcal{IR}_{\mathcal{B}}$. Let j be an arbitrary element of $\{1, \ldots, \ell+1\}$. It is clear that the pair $\left(\varphi^* + \epsilon^*_j\varphi^j, \theta^j\right)$ satisfies (4.7). Recall (4.3) and (4.14). Since $\left(\varphi^* + \epsilon^*_j\varphi^j\right) \in \mathcal{IR}'_{\mathcal{B}}$ we have $G^{-1}_{\bullet\mathcal{B}}\left(q + (\triangle Q)_{\bullet L}\left(\varphi^* + \epsilon^*_j\varphi^j\right)\right) \geq 0$. This implies that $G^{-1}_{\bullet\mathcal{B}}\left(q + \triangle Q\theta^j\right) \geq 0$ and thus $\theta^j \in \mathcal{IR}_{\mathcal{B}}$.

Case 2: $i \in \{\ell+2, \ldots, k+1\}$

Since $\theta^* \in relint(S_\theta)$, for each $i \in \{\ell+2, \ldots, k+1\}$ there exists a $\tilde{\epsilon}_i > 0$ such that $\begin{bmatrix} \theta^*_{L\bullet} \\ \theta^*_{\Lambda\bullet} \end{bmatrix} + \tilde{\delta}_i \begin{bmatrix} -\tilde{Q}\chi^{i-\ell-1} \\ \chi^{i-\ell-1} \end{bmatrix} \in S_\theta$ for all $\tilde{\delta}_i \in [0, \tilde{\epsilon}_i]$. Thus, for each $i \in \{\ell+2, \ldots, k+1\}$, define θ^i as:

$$\theta^i = \begin{bmatrix} \theta^i_{L\bullet} \\ \theta^i_{\Lambda\bullet} \end{bmatrix} = \begin{bmatrix} \varphi^* - \tilde{Q}\left(\chi^* + \tilde{\epsilon}_i \chi^{i-\ell-1}\right) \\ \chi^* + \tilde{\epsilon}_j \chi^{i-\ell-1} \end{bmatrix} \tag{4.16}$$

We now show that for each $i \in \{\ell+2, \ldots, k+1\}$, $\theta^i \in \mathcal{IR}_{\mathscr{B}}$. Notice that the pair $\left(\varphi^*, \theta^j\right)$ satisfies (4.7) for all $i \in \{\ell+2, \ldots, k+1\}$. Recall (4.3) and (4.14). Since $\varphi^* \in \mathcal{IR}'_{\mathscr{B}}$ it is clear that $G^{-1}_{\bullet\mathscr{B}}\left(q + (\triangle Q)_{\bullet L}\,\varphi^*\right) \geq 0$. Therefore, for all $i \in \{\ell+2, \ldots, k+1\}$, $G^{-1}_{\bullet\mathscr{B}}\left(q + \triangle Q\theta^i\right) \geq 0$ and thus $\theta^i \in \mathcal{IR}_{\mathscr{B}}$.

Cases 1 and 2 have now been completed. We next show the affine independence of $\theta^1, \ldots, \theta^{k+1}$. To accomplish this we show that the vectors $\theta^2 - \theta^1, \ldots, \theta^{k+1} - \theta^1$ are linearly independent. From (4.15) and (4.16) observe:

$$\begin{bmatrix} \theta^i_{L\bullet} - \theta^1_{L\bullet} \\ \theta^i_{\Lambda\bullet} - \theta^1_{\Lambda\bullet} \end{bmatrix} = \begin{cases} \begin{bmatrix} \epsilon^*_i \varphi^i - \epsilon^*_1 \varphi^1 \\ 0 \end{bmatrix} & \text{for } i \in \{2, \ldots, \ell+1\} \\ \begin{bmatrix} -\tilde{Q}\tilde{\epsilon}_i \chi^{i-\ell-1} - \epsilon^*_i \varphi^1 \\ \tilde{\epsilon}_i \chi^{i-\ell-1} \end{bmatrix} & \text{for } i \in \{\ell+2, \ldots, k+1\} \end{cases} \tag{4.17}$$

Since $\varphi^1, \ldots, \varphi^{\ell+1}$ are affinely independent, the vectors $\varphi^2 - \varphi^1, \ldots, \varphi^{\ell+1} - \varphi^1$ are linearly independent (and thus are nonzero). Furthermore, since $\epsilon^*_i > 0$ for all $i \in \{1, \ldots, \ell+1\}$, $\epsilon^*_i \varphi^i - \epsilon^*_1 \varphi^1 \neq 0$ for all $i \in \{2, \ldots, \ell+1\}$. This shows that $\theta^i - \theta^1 \neq 0$ for all $i \in \{2, \ldots, \ell+1\}$. Also notice that $\theta^i - \theta^1 \neq 0$ for all $i \in \{\ell+2, \ldots, k+1\}$ since $\chi^1, \ldots, \chi^{k-\ell}$ are linearly independent (and are thus nonzero).

Now, for each pair of distinct $i, j \in \{2, \ldots, \ell+1\}$ it is clear from (4.17) that $\theta^i - \theta^1$ is linearly independent from $\theta^j - \theta^1$ since $\varphi^1, \ldots, \varphi^{\ell+1}$ are affinely independent. It is also clear for each pair of distinct $i, j \in \{\ell+2, \ldots, k+1\}$ that $\theta^i - \theta^1$ is linearly independent from $\theta^j - \theta^1$ since $\chi^1, \ldots, \chi^{k-\ell}$ are linearly independent. Finally, for each pair (i, j) such that $i \in \{2, \ldots, \ell+1\}$ and $j \in \{\ell+2, \ldots, k+1\}$, it is clear that $\theta^i - \theta^1$ is linearly independent from $\theta^j - \theta^1$ since $\chi^1, \ldots, \chi^{k-\ell}$ are linearly independent (and are thus nonzero). Thus, the points $\theta^1, \ldots, \theta^{k+1} \in \mathcal{IR}_{\mathscr{B}}$ are affinely independent, and hence, $\mathcal{IR}_{\mathscr{B}}$ is full dimensional.

($\Rightarrow$): We proceed with this direction of the proof by proving the contrapositive. As discussed previously, for any $\theta \in \mathcal{IR}_{\mathscr{B}}$, there exists $\varphi \in \mathcal{IR}'_{\mathscr{B}}$ and $\chi \in \mathbb{R}^{k-\ell}$ such that $\begin{bmatrix} \theta_{L\bullet} \\ \theta_{\Lambda\bullet} \end{bmatrix} = \begin{bmatrix} \varphi - \tilde{Q}\chi \\ \chi \end{bmatrix}$. The maximum number of linearly independent vectors that can be found in $\mathcal{IR}_{\mathscr{B}}$ is bounded above by the sum of the maximum number of linearly independent vectors in $\mathcal{IR}'_{\mathscr{B}}$ and the maximum number of linearly independent vectors in $\mathbb{R}^{k-\ell}$. Since the maximum number of linearly independent vectors in $\mathbb{R}^{k-\ell}$ is clearly $k-\ell$, we observe that if the maximum number of linearly independent vectors in $\mathcal{IR}'_{\mathscr{B}}$ is less than ℓ, the maximum number of linearly independent vectors in $\mathcal{IR}_{\mathscr{B}}$ is less than $k-\ell+\ell=k$. Consequently, $\mathcal{IR}_{\mathscr{B}}$ cannot be full dimensional unless $\mathcal{IR}'_{\mathscr{B}}$ is full dimensional. □

Proposition 4.24. Two invariancy regions $\mathcal{IR}_{\mathscr{B}_1}$ and $\mathcal{IR}_{\mathscr{B}_2}$ are adjacent if and only if $\mathcal{IR}'_{\mathscr{B}_1}$ and $\mathcal{IR}'_{\mathscr{B}_2}$ are adjacent.

Proof. The same arguments used to prove Proposition 3.3 can be used to show that $dim(\mathcal{IR}_{\mathscr{B}_1} \cap \mathcal{IR}_{\mathscr{B}_2}) = k-1$ if and only if $dim\left(\mathcal{IR}'_{\mathscr{B}_1} \cap \mathcal{IR}'_{\mathscr{B}_2}\right) = \ell - 1$. The desired result follows. □

Together, Propositions 3.3 and 3.4 show that if one desires to solve an mpLCP in which $\triangle Q$ is not of full column rank, one can instead solve mpLCP$'$. The resulting partition of S_φ can be directly used to obtain the desired partition of S_θ. Thus, we assume throughout this work that $\triangle Q$ is of full column rank.

4.3 Phase 2: Partitioning the parameter space

In this section we introduce the theory necessary for developing an algorithm that can be used to partition S_θ, given an initial basis $\mathscr{B}_0$ such that $dim\left(\mathcal{IR}_{\mathscr{B}_0}\right) \geq k-1$. We present the algorithm for partitioning S_θ in Section 4.5.

To solve an instance of mpLCP we first need a method so that given a complementary feasible basis $\mathscr{B}$, we can determine a complementary feasible basis $\mathscr{B}'$ which is adjacent to $\mathscr{B}$. Observe the following proposition.

Proposition 4.25. [18] If $M \in \mathbb{R}^{h \times h}$ is column sufficient and two bases $\mathscr{B}_1$ and $\mathscr{B}_2$ are adjacent, then $|\mathscr{B}_1 \cap \mathscr{B}_2| \geq h-2$.

Proof. Proposition 4.25 is proved in [18] (Lemma 3.8, pp. 6). □

The result of Proposition 4.25 is quite powerful as it implies that given any basis $\mathscr{B}$, all complementary bases which are adjacent to $\mathscr{B}$ can be obtained by replacing either 1 or 2 elements of $\mathscr{B}$ with their complements.

Definition 4.26. Replacing a single element of a basis with its complement is referred to as a *diagonal pivot.*

Definition 4.27. Replacing two elements of a basis with their complements is referred to as an *exchange pivot.*

These terms arise from techniques for solving LCP which consider the LCP in a tableau format and rely on principal pivoting to find feasible solutions. For a given basis $\mathcal{B}$, the corresponding tableau is the augmented matrix

$$T_{\mathcal{B}} := \left[\; G_{\bullet\mathcal{B}}^{-1} G \;\vdots\; G_{\bullet\mathcal{B}}^{-1} \left(q + \triangle Q\theta\right) \;\right] \tag{4.18}$$

where the right hand side (RHS) is precisely $\nu_{\mathcal{B}}(\theta)$.

It is important to note that given a basis $\mathcal{B}$, not every diagonal or exchange pivot results in a new basis. To see this, suppose that in a particular diagonal or exchange pivot $J \subset \mathcal{B}$ is the set of indices replaced with their complements. If $rank\left(G_{\bullet((B\setminus J)\cup\overline{J})}\right) \neq h$ then $\left((B \setminus J) \cup \overline{J}\right)$ cannot be a basis. Additionally, even if a pivot on $\mathcal{B}$ does result in a new basis $\mathcal{B}'$, the bases $\mathcal{B}$ and $\mathcal{B}'$ may not be adjacent. Due to these facts, we next need to determine conditions under which pivots will yield new adjacent feasible bases. Such conditions can be developed using the tableau $T_{\mathcal{B}}$.

We first consider diagonal pivots. Since principal pivoting has been studied extensively in the context of LCP, the following result is well known in the literature.

Observation 4.28. Given a complementary feasible basis $\mathcal{B}$ and any index $i \in \mathcal{B}$, the set $(\mathcal{B} \setminus \{i\}) \cup \{\overline{\imath}\}$ is a basis if and only if $(T_{\mathcal{B}})_{i,\overline{\imath}} \neq 0$.

The following proposition and its corollary are slightly modified from [18].

Proposition 4.29. Given a complementary feasible basis $\mathcal{B}$, suppose that for some index $i \in \mathcal{B}$ the set $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\overline{\imath}\}$ is a basis. Then $\mathcal{B}$ and $\mathcal{B}'$ are adjacent.

Proof. The proposition is implied by the facts that $\mathcal{C}(\mathcal{B}) \cap \mathcal{C}(\mathcal{B}') = cone\left(G_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ and $dim\left(cone\left(G_{\bullet(\mathcal{B}\setminus\{i\})}\right)\right) = h - 1$. Therefore, by Definition 3.23, the bases $\mathcal{B}$ and $\mathcal{B}'$ are adjacent. □

Corollary 4.30. Given a complementary feasible basis $\mathcal{B}$, suppose that for some index $i \in \mathcal{B}$ the set $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\overline{\imath}\}$ is a basis. If M is column sufficient then $\mathcal{C}(\mathcal{B}')$ is the unique complementary cone adjacent to $\mathcal{C}(\mathcal{B})$ along the facet $cone\left(G_{\bullet(\mathcal{B}\setminus\{i\})}\right)$.

Proof. The Corollary is implied by the fact that for column sufficient matrices the relative interiors of all complementary cones are disjoint. □

Together Observation 4.28 and Proposition 4.29 provide conditions under which an adjacent complementary feasible basis $\mathcal{B}'$ can be derived from a given complementary feasible basis $\mathcal{B}$ by using a single diagonal pivot.

We now move our attention to exchange pivots and present the following two new propositions.

Proposition 4.31. For a given complementary basis $\mathcal{B}$, suppose there exist distinct $i, j \in \mathcal{B}$ such that $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a complementary basis which is adjacent to $\mathcal{B}$. Then if M is column sufficient, either $(T_{\mathcal{B}})_{i,\bar{\imath}} = 0$ or $(T_{\mathcal{B}})_{j,\bar{\jmath}} = 0$.

Proof. Assume without loss of generality that $(T_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$. Then $\hat{\mathcal{B}} = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a complementary basis and $\mathcal{C}(\mathcal{B}) \cap \mathcal{C}(\hat{\mathcal{B}}) = cone\left(G_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Furthermore, by Corollary 4.30, $\hat{\mathcal{B}}$ is the unique basis whose complementary cone intersects $\mathcal{C}(\mathcal{B})$ along this facet. Therefore, since $\mathcal{B}'$ is also adjacent to $\mathcal{B}$, it must be that $\mathcal{C}(\mathcal{B}) \cap \mathcal{C}(\mathcal{B}') \subseteq cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right)$. Hence, there must exist $J' \subset \mathcal{B}'$ such that $|J'| = h-1$ and $dim\left(cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right) \cap cone\left(G_{\bullet J'}\right)\right) = h-1$. Note that since $|J'| = h-1$, either $\bar{\imath}$ or $\bar{\jmath}$ must be a member of J'. Consider the following two cases:

Case 1: $\bar{\imath} \in J'$

Since $dim\left(cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right) \cap cone\left(G_{\bullet J'}\right)\right) = h-1$, we have $G_{\bullet\bar{\imath}} \in span\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right)$. Now, suppose that $G_{\bullet\bar{\imath}} \notin cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right)$. Then $\mathcal{C}(\mathcal{B})$ and $\mathcal{C}(\mathcal{B}')$ can only be adjacent along the facet $cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right)$ if $\mathcal{C}(\mathcal{B}) = cone\left(G_{\bullet\mathcal{B}}\right) \subset cone\left(G_{\bullet\left((\mathcal{B}\setminus\{i\})\cup\{\bar{\imath}\}\right)}\right) = \mathcal{C}(\hat{\mathcal{B}})$, but this contradicts the fact that M is column sufficient since in this case $\mathcal{C}(\hat{\mathcal{B}})$ and $\mathcal{C}(\mathcal{B})$ do not have disjoint relative interiors. Suppose instead that $G_{\bullet\bar{\imath}} \in cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right)$. This implies that $\mathcal{C}(\hat{\mathcal{B}}) = cone\left(G_{\bullet\left((\mathcal{B}\setminus\{i\})\cup\{\bar{\imath}\}\right)}\right) \subseteq cone\left(G_{\bullet\mathcal{B}}\right) = \mathcal{C}(\mathcal{B})$, which means that one of the following must hold: (i) $\mathcal{C}(\hat{\mathcal{B}})$ is not full dimensional, or (ii) $\mathcal{C}(\hat{\mathcal{B}})$ and $\mathcal{C}(\mathcal{B})$ do not have disjoint relative interiors. Notice, however, that the first of these contradicts the fact that $\hat{\mathcal{B}}$ is a basis, and the latter contradicts the fact that M is column sufficient.

Case 2: $\bar{\jmath} \in J'$

Since $dim\left(cone\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right) \cap cone\left(G_{\bullet J'}\right)\right) = h-1$, we have $G_{\bullet\bar{\jmath}} \in span\left(G_{\bullet(\mathcal{B}\setminus\{j\})}\right)$. Since $T_{\mathcal{B}} = \left[\; G_{\bullet\mathcal{B}}^{-1}G \;\vdots\; G_{\bullet\mathcal{B}}^{-1}\left(q + \triangle Q\theta\right) \;\right]$, we have $(T_{\mathcal{B}})_{j,\gamma} = 0$ for all $\gamma \in (\mathcal{B} \setminus \{j\})$. Thus, since $G_{\bullet\bar{\jmath}}$ is a linear combination of the columns of $G_{\bullet(\mathcal{B}\setminus\{j\})}$, it must be that $(T_{\mathcal{B}})_{j,\bar{\jmath}} = 0$.

Cases 1 and 2 are now complete. The contradictions found in Case 1 show that either $\bar\imath \notin J'$ or $(T_{\mathcal{B}})_{i,\bar\imath} = 0$. Clearly, if $(T_{\mathcal{B}})_{i,\bar\imath} = 0$ the thesis of the proposition holds. If, on the other hand, $\bar\imath \notin J'$ then $\bar\jmath \in J'$ which, as is shown in Case 2, implies that $(T_{\mathcal{B}})_{j,\bar\jmath} = 0$. Thus, the claim of the proposition holds in all cases. □

Although the claim of the following proposition is different from those of Columbano et al. [18], aspects of one of the proofs in their work are used in our proof.

Proposition 4.32. Let a complementary basis $\mathcal{B}$ and distinct $i, j \in \mathcal{B}$ be given. The set $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar\imath, \bar\jmath\}$ is a complementary basis such that $\mathcal{C}(\mathcal{B})$ is adjacent to $\mathcal{C}(\mathcal{B}')$ along the facet $cone\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ if and only if $(T_{\mathcal{B}})_{i,\bar\imath} = 0$, $(T_{\mathcal{B}})_{j,\bar\imath} > 0$, and $(T_{\mathcal{B}})_{i,\bar\jmath} \neq 0$.

Proof. ($\Leftarrow$): We first show that $\mathcal{B}'$ is a basis and then show that $\mathcal{C}(\mathcal{B})$ is adjacent to $\mathcal{C}(\mathcal{B}')$ along the facet $cone\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Since $T_{\mathcal{B}} = \left[\ G^{-1}_{\bullet\mathcal{B}} G \;\vdots\; G^{-1}_{\bullet\mathcal{B}} (q + \triangle Q\theta)\ \right]$, we have $(T_{\mathcal{B}})_{i,\gamma} = 0$ for all $\gamma \in \mathcal{B} \setminus \{i\}$ and $(T_{\mathcal{B}})_{j,\xi} = 0$ for all $\xi \in \mathcal{B} \setminus \{j\}$. Thus, since $(T_{\mathcal{B}})_{j,\bar\imath} > 0$, $G_{\bullet\bar\imath}$ cannot be a linear combination of the columns of $G_{\bullet(\mathcal{B}\setminus\{i,j\})}$. Furthermore, since $(T_{\mathcal{B}})_{i,\bar\imath} = 0$ and $(T_{\mathcal{B}})_{i,\bar\jmath} \neq 0$, $G_{\bullet\bar\jmath}$ cannot be a linear combination of the columns of $G_{\bullet((\mathcal{B}\setminus\{i,j\})\cup\{\bar\imath\})}$. Thus, the columns of $G_{\bullet\mathcal{B}'}$ are linearly independent, showing that $\mathcal{B}'$ is a basis.

We now show that $\mathcal{C}(\mathcal{B})$ is adjacent to $\mathcal{C}(\mathcal{B}')$ along the facet $cone\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Notice that for any set $J \subset \mathcal{E}$, $q + \triangle Q\theta$ lies in the relative interior of $cone\left(G_{\bullet J}\right)$ if and only if $q + \triangle Q\theta$ is a strictly positive combination of the columns of $G_{\bullet J}$, i.e., for each $\gamma \in J$ there exists $\beta_\gamma > 0$ such that $q + \triangle Q\theta = \sum\limits_{\gamma\in J} \beta_\gamma G_{\bullet\gamma}$. Thus, consider

$$\begin{aligned}\mathcal{W}(\beta) :&= \sum_{\gamma\in(\mathcal{B}\setminus\{i\})} \beta_\gamma G_{\bullet\gamma} \\ &= \beta_j G_{\bullet j} + \sum_{\gamma\in(\mathcal{B}\setminus\{i,j\})} \beta_\gamma G_{\bullet\gamma} \qquad (4.19)\end{aligned}$$

Recall that $(T_{\mathcal{B}})_{\bullet\bar\imath} = G^{-1}_{\bullet\mathcal{B}} G_{\bullet\bar\imath}$, which implies:

$$\begin{aligned}G_{\bullet\bar\imath} &= G_{\bullet\mathcal{B}} (T_{\mathcal{B}})_{\bullet\bar\imath} \\ &= \sum_{\gamma\in\mathcal{B}} G_{\bullet\gamma} (T_{\mathcal{B}})_{\gamma\bar\imath} \\ &= \sum_{\gamma\in(\mathcal{B}\setminus\{i\})} G_{\bullet\gamma} (T_{\mathcal{B}})_{\gamma\bar\imath}\end{aligned}$$

$$= G_{\bullet j}(T_{\mathcal{B}})_{j,\bar{\imath}} + \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} G_{\bullet\gamma}(T_{\mathcal{B}})_{\gamma\bar{\imath}} \tag{4.20}$$

Since $(T_{\mathcal{B}})_{j,\overline{\imath}} > 0$, (4.20) gives $G_{\bullet j} = \dfrac{1}{(T_{\mathcal{B}})_{j,\overline{\imath}}} G_{\bullet\overline{\imath}} - \displaystyle\sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} G_{\bullet\gamma} \dfrac{(T_{\mathcal{B}})_{\gamma\overline{\imath}}}{(T_{\mathcal{B}})_{j,\overline{\imath}}}$. Using this result and substituting into (4.19) yields

$$\begin{aligned} \mathcal{W}(\beta) &= \beta_j \left(\frac{1}{(T_{\mathcal{B}})_{j,\bar{\imath}}} G_{\bullet\bar{\imath}} - \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} G_{\bullet\gamma} \frac{(T_{\mathcal{B}})_{\gamma\bar{\imath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}} \right) + \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} \beta_\gamma G_{\bullet\gamma} \\ &= \frac{\beta_j}{(T_{\mathcal{B}})_{j,\bar{\imath}}} G_{\bullet\bar{\imath}} - \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} \left(\beta_\gamma - \beta_j \frac{(T_{\mathcal{B}})_{\gamma\bar{\imath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}} \right) G_{\bullet\gamma} \end{aligned} \tag{4.21}$$

From (4.19) and (4.21) we observe that by selecting $\tilde{\beta}$ so that: (i) $\tilde{\beta}_\gamma > 0$ for all $\gamma \in (\mathcal{B} \setminus \{i\})$, and (ii) $\tilde{\beta}_\gamma > \tilde{\beta}_j \dfrac{(T_{\mathcal{B}})_{\gamma\overline{\imath}}}{(T_{\mathcal{B}})_{j,\overline{\imath}}}$ for all $\gamma \in (\mathcal{B} \setminus \{i,j\})$, we have that $\mathcal{W}(\tilde{\beta})$ is in the relative interior of both $\mathit{cone}\left(G_{\bullet\,(\mathcal{B}\setminus\{i\})}\right)$ and $\mathit{cone}\left(G_{\bullet\,(\mathcal{B}'\setminus\{\overline{\jmath}\})}\right)$. This shows that $\mathit{dim}\left(\mathit{cone}\left(G_{\bullet\,(\mathcal{B}\setminus\{i\})}\right) \cap \mathit{cone}\left(G_{\bullet\,(\mathcal{B}'\setminus\{\overline{\jmath}\})}\right)\right) = h - 1$ and therefore $\mathcal{B}$ and $\mathcal{B}'$ are adjacent.

($\Rightarrow$): We prove this direction by contradiction. Consider the following 3 cases:

Case 1: $(T_{\mathcal{B}})_{i,\overline{\imath}} \neq 0$

By Corollary 4.30, $\hat{\mathcal{B}} = (\mathcal{B} \setminus \{i\}) \cup \{\overline{\imath}\}$ is the unique basis such that $\mathcal{C}(\hat{\mathcal{B}})$ is adjacent to $\mathcal{C}(\mathcal{B})$ along the facet $\mathit{cone}\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. This is a contradiction.

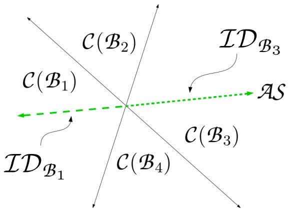
Case 2: $(T_{\mathcal{B}})_{j,\overline{\imath}} \leq 0$

Since $\mathcal{B}'$ is a basis, the unique way to represent $\mathcal{W}(\beta)$ as a linear combination of the columns of $G_{\bullet\mathcal{B}'}$ is (4.21). Therefore, in this case there does not exist β such that $\mathcal{W}(\beta)$ lies in both the relative interiors of $\mathit{cone}\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ and $\mathcal{C}(\mathcal{B}')$. Hence, $\mathcal{C}(\mathcal{B})$ cannot be adjacent to $\mathcal{C}(\mathcal{B}')$ along $\mathit{cone}\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$, which is a contradiction.

Case 3: $(T_{\mathcal{B}})_{i,\overline{\jmath}} = 0$

Since $(T_{\mathcal{B}})_{i,\overline{\imath}} = 0$, the matrix $G^{-1}_{\bullet\mathcal{B}} G_{\bullet\mathcal{B}'}$ has a row of all zeros. Thus $G^{-1}_{\bullet\mathcal{B}} G_{\bullet\mathcal{B}'}$ is not invertible, which is a contradiction since both $\mathcal{B}$ and $\mathcal{B}'$ are bases.

Finding a contradiction in each of the cases above shows that we must have $(T_{\mathcal{B}})_{i,\overline{\imath}} = 0$, $(T_{\mathcal{B}})_{j,\overline{\imath}} > 0$, and $(T_{\mathcal{B}})_{i,\overline{\jmath}} \neq 0$. □



$\mathcal{C}(\mathcal{B}_2)$
$\mathcal{ID}_{\mathcal{B}_3}$
$\mathcal{C}(\mathcal{B}_1)$
$\mathcal{AS}$
$\mathcal{C}(\mathcal{B}_3)$
$\mathcal{ID}_{\mathcal{B}_1}$
$\mathcal{C}(\mathcal{B}_4)$

Figure 4.1: Consider the four complementary cones $\mathcal{C}(\mathcal{B}_1)$, $\mathcal{C}(\mathcal{B}_2)$, $\mathcal{C}(\mathcal{B}_3)$ and $\mathcal{C}(\mathcal{B}_4)$ and the two invariant domains $\mathcal{ID}_{\mathcal{B}_1}$ and $\mathcal{ID}_{\mathcal{B}_3}$. The invariant domains $\mathcal{ID}_{\mathcal{B}_1}$ and $\mathcal{ID}_{\mathcal{B}_3}$ map to adjacent invariancy regions $\mathcal{IR}_{\mathcal{B}_1}$ and $\mathcal{IR}_{\mathcal{B}_3}$ even though the complementary cones $\mathcal{C}(\mathcal{B}_1)$ and $\mathcal{C}(\mathcal{B}_3)$ are not adjacent.

By combining the results of Propositions 4.25, 4.29, 4.31 and 4.32 as well as Observation 4.28 and Corollary 4.30, we are now able to develop the following strategy for finding all complementary bases which are adjacent to a given basis $\mathcal{B}$:

1. Calculate the tableau $T_{\mathcal{B}}$ associated with basis $\mathcal{B}$.

2. For any $i \in \mathcal{B}$ such that $(T_{\mathcal{B}})_{i,\overline{\imath}} \neq 0$, the set $(\mathcal{B} \setminus \{i\}) \cup \{\overline{\imath}\}$ is a complementary basis adjacent to $\mathcal{B}$.

3. For any distinct $i, j \in \mathcal{B}$ such that $(T_{\mathcal{B}})_{i,\overline{\imath}} = 0$, $(T_{\mathcal{B}})_{j,\overline{\imath}} > 0$ and $(T_{\mathcal{B}})_{i,\overline{\jmath}} \neq 0$, the set $(\mathcal{B} \setminus \{i, j\}) \cup \{\overline{\imath}, \overline{\jmath}\}$ is a feasible complementary basis adjacent to $\mathcal{B}$.

Recall, however, that our goal is to partition S_θ. Although there is a strong relationship between complementary cones and invariancy regions, it is not always the case that adjacent invariancy regions result from adjacent complementary cones. To see this, observe Figure 4.1.

It has been proved by Columbano et al. [18] (Thm 5.10 pp. 21), though, that for any pair of adjacent full dimensional invariancy regions $\mathcal{IR}_{\mathcal{B}_i}$ and $\mathcal{IR}_{\mathcal{B}_j}$ with $i < j$, there always exists a sequence of invariancy regions $\{\mathcal{IR}_{\mathcal{B}_\ell}\}_{\ell=i+1}^{j-1}$ such that $\mathcal{C}(\mathcal{B}_\gamma)$ and $\mathcal{C}(\mathcal{B}_{\gamma+1})$ are adjacent for all $\gamma \in \{i, \ldots, j-1\}$ and $dim\,(\mathcal{C}(\mathcal{B}_\gamma) \cap \mathcal{AS}) = k-1$ for all $\gamma \in \{i+1, \ldots, j-1\}$. In general, the method for finding one of these sequences introduced by Columbano et al. [18] relies on an ϵ-perturbation of the affine subspace $\mathcal{AS}$. In order to obtain a lower worst-case complexity, we derive an alternative method in which we determine the sequences directly. We first present a brief outline of this method and then provide the details necessary for its implementation.

Outline 4.33:

Preliminaries: Let $\mathscr{R}$ represent a set of invariancy regions for which adjacent regions need to be found. Let $\mathscr{K}$ represent the set of feasible bases discovered.

Phase 1 – Initialization: Find an initial full dimensional invariancy region $\mathcal{IR}_{\mathscr{B}_0}$ and add it to $\mathscr{R}$. Add $\mathscr{B}_0$ to $\mathscr{K}$.

Phase 2 – Main Step:

1. Select $\mathcal{IR}_{\mathscr{B}} \in \mathscr{R}$ and remove it from $\mathscr{R}$.
2. Determine the set of feasible bases which have invariancy regions that are: (i) adjacent to $\mathcal{IR}_{\mathscr{B}}$, and (ii) at least $(k-1)$–dimensional.
3. For each basis $\hat{\mathscr{B}}$ found in step 2, if $\hat{\mathscr{B}} \notin \mathscr{K}$, add $\mathcal{IR}_{\hat{\mathscr{B}}}$ to $\mathscr{R}$ and $\hat{\mathscr{B}}$ to $\mathscr{K}$.
4. If $\mathscr{R} = \emptyset$, STOP. Otherwise, go back to step 1.

Note that in Outline 4.33 the main step serves as a basis for Algorithm 4.1, while the initialization phase serves as a basis for Algorithm 4.2. Both of these algorithms are presented in Section 4.5. We now discuss the details necessary to implement the main steps of the procedure above. The details of the initialization step, i.e., determining an initial feasible basis $\mathscr{B}_0$ such that $dim\left(\mathcal{IR}_{\mathscr{B}_0}\right) \geq k-1$, are given in Section 4.4. For any feasible complementary basis $\mathscr{B}$ and index $i \in \mathscr{B}$, we define the following sets which will be useful during the remainder of this discussion:

$$\mathrm{Z}_{\mathscr{B}} := \left\{ j \in \mathscr{B} : \left(G^{-1}_{\bullet\mathscr{B}}\right)_{j\bullet}(q + \triangle Q\theta) = 0 \quad \forall\, \theta \in S_\theta \right\} \tag{4.22}$$

$$\mathrm{H}^i_{\mathscr{B}} := \left\{ j \in \mathscr{B} \setminus \{i\} : \exists\, \beta \neq 0 \text{ s.t. } \left(\left(G^{-1}_{\bullet\mathscr{B}}\right)_{j\bullet} - \beta \left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet} \right)(q + \triangle Q\theta) = 0 \;\; \forall\, \theta \in S_\theta \right\} \tag{4.23}$$

Here $\mathrm{Z}_{\mathscr{B}}$ is the set of indices in $\mathscr{B}$ for which the RHS of $T_{\mathscr{B}}$ is zero. For a given $\theta \in S_\theta$ these RHS values can be interpreted as the multipliers on the columns of $G_{\bullet\mathscr{B}}$ needed to represent $q + \triangle Q\theta$ as a linear combination of the columns of G. Thus, if the RHS value is zero for some index i, this indicates that the column $G_{\bullet i}$ is unnecessary in the representation of $q + \triangle Q\theta$. There is also another interpretation. Notice from (4.3) that for each $i \in \mathscr{B}$, $\left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet}(q + \triangle Q\theta) \geq 0$ is a defining inequality of $\mathcal{IR}_{\mathscr{B}}$. Thus if there is some $i \in \mathscr{B}$ for which the RHS of $T_{\mathscr{B}}$ is zero, the associated defining inequality of $\mathcal{IR}_{\mathscr{B}}$ is $0 \geq 0$, which is trivially satisfied. Now consider $\mathrm{H}^i_{\mathscr{B}}$. Given an index $i \in \mathscr{B}$, the set $\mathrm{H}^i_{\mathscr{B}}$ is the set of indices in $\mathscr{B} \setminus \{i\}$ for which the RHS of $T_{\mathscr{B}}$ is a constant multiple of the

RHS of $T_{\mathscr{B}}$ associated with index i. Thus, for all $j \in \mathrm{H}^i_{\mathscr{B}}$, the defining constraints of $\mathcal{IR}_{\mathscr{B}}$ associated with i and j are implied by the same hyperplane. For an index $i \in \mathscr{B}$, we use $\hbar^i_{\mathscr{B}}$ to denote the hyperplane which forms the defining constraint of $\mathcal{IR}_{\mathscr{B}}$ associated with i. Thus, we have:

$$\hbar^i_{\mathscr{B}} := \left\{\theta \in \mathbb{R}^k : \left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet} (q + \triangle Q\theta) = 0\right\}. \tag{4.24}$$

We now introduce several propositions whose results allow us to perform the steps of Outline 4.33. Each of these propositions introduces a linear program (LP) that can be solved in order to determine, for example, if a given invariancy region is full dimensional. We note that these LPs do not need to be solved to optimality, but rather a feasible solution must be found which has an associated objective function value which is strictly positive. To aid in understanding of the details presented throughout the following discussion, we provide here two instances of mpLCP. As we develop the theory necessary for partitioning S_θ, we also show directly how this theory can be applied to each instance. Observe the following examples:

Example 1

$$w - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} z = \begin{bmatrix} 6 \\ -1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & -1 \\ 1 & -3 \end{bmatrix} \theta$$
$$w^\top z = 0$$
$$w, z \geq 0$$
$$S_\theta = [-2, 2]^2$$

Example 2

$$w - \begin{bmatrix} 5 & 4 & 0 \\ 4 & 5 & 0 \\ -1 & 0 & 1 \end{bmatrix} z = \begin{bmatrix} -3 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ -1 & 3 \\ 0 & 0 \end{bmatrix} \theta$$
$$w^\top z = 0$$
$$w, z \geq 0$$
$$S_\theta = [-2, 2]^2$$

For the sake of clarity, when discussing the examples above we use variable names to describe the elements of each basis rather than their corresponding indices. We now provide the solution to each example. How these solutions are determined is shown during the discussion that follows.

Solution 1

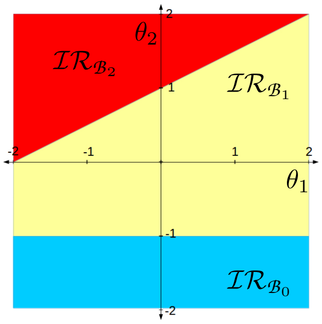


Figure 4.2: Partition of S_θ - Example 1

Solution 2

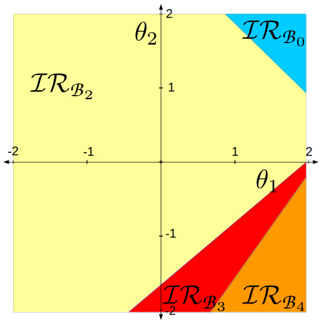


Figure 4.3: Partition of S_θ - Example 2

Invariancy regions for Example 1

$$\mathcal{IR}_{\mathcal{B}_0} = \left\{ \theta \in S_\theta : \begin{array}{c} w_1 = \theta_1 + 6 \geq 0 \\ w_2 = -\theta_2 - 1 \geq 0 \\ w_3 = \theta_1 - 3\theta_2 + 1 \geq 0 \end{array} \right\}, \quad \mathcal{IR}_{\mathcal{B}_1} = \left\{ \theta \in S_\theta : \begin{array}{c} w_1 = \theta_1 + 6 \geq 0 \\ z_2 = \theta_2 + 1 \geq 0 \\ w_3 = \theta_1 - 2\theta_2 + 2 \geq 0 \end{array} \right\}$$

$$\mathcal{IR}_{\mathcal{B}_2} = \left\{ \theta \in S_\theta : \begin{array}{c} w_1 = \theta_1 + 6 \geq 0 \\ w_2 = -\theta_1 + 2\theta_2 - 2 \geq 0 \\ z_3 = -\theta_1 + 3\theta_2 - 1 \geq 0 \end{array} \right\}$$

Invariancy regions for Example 2

$$\mathcal{IR}_{\mathcal{B}_0} = \left\{ \theta \in S_\theta : \begin{array}{c} w_1 = \theta_1 + \theta_2 - 3 \geq 0 \\ w_2 = -\theta_1 + 3\theta_2 + 1 \geq 0 \\ w_3 = 0 \end{array} \right\}, \quad \mathcal{IR}_{\mathcal{B}_2} = \left\{ \theta \in S_\theta : \begin{array}{c} z_1 = -\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 + \frac{3}{5} \geq 0 \\ w_2 = -\frac{9}{5}\theta_1 + \frac{11}{5}\theta_2 + \frac{17}{5} \geq 0 \\ z_3 = -\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 + \frac{3}{5} \geq 0 \end{array} \right\}$$

$$\mathcal{IR}_{\mathcal{B}_3} = \left\{ \theta \in S_\theta : \begin{array}{c} z_1 = -\theta_1 + \frac{7}{9}\theta_2 + \frac{19}{9} \geq 0 \\ z_2 = \theta_1 - \frac{11}{9}\theta_2 - \frac{17}{9} \geq 0 \\ z_3 = -\theta_1 + \frac{7}{9}\theta_2 + \frac{19}{9} \geq 0 \end{array} \right\}, \quad \mathcal{IR}_{\mathcal{B}_4} = \left\{ \theta \in S_\theta : \begin{array}{c} w_1 = \frac{9}{5}\theta_1 - \frac{7}{5}\theta_2 - \frac{19}{5} \geq 0 \\ z_2 = \frac{1}{5}\theta_1 - \frac{3}{5}\theta_2 - \frac{1}{5} \geq 0 \\ z_3 = 0 \end{array} \right\}$$

For each of the above examples we claim that an initial basis $\mathcal{B}_0 = \{w_1, w_2, w_3\}$ is feasible. A discussion on obtaining initial bases is provided in Section 4.4. The following proposition provides the tools necessary for determining whether or not the invariancy region associated with a given basis is full dimensional.

Proposition 4.34. For a given feasible complementary basis $\mathcal{B}$, $\mathcal{IR}_{\mathcal{B}}$ is full dimensional if and only if $|\mathrm{Z}_{\mathcal{B}}| \leq h - k$ and there exists $\hat{\theta} \in S_\theta$ such that $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\hat{\theta}) > 0$ for all $i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}}$, i.e., if the following LP has a strictly positive optimal value:

$$\begin{array}{rll} LP_D(\mathcal{B}) := \max\limits_{\lambda,\theta} & \lambda & \\ \text{s.t.} & \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(\triangle Q\theta) - \lambda\mathbf{1} \geq -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q & \forall\, i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}} \\ & \theta \in S_\theta & \end{array} \tag{4.25}$$

Proof. ($\Leftarrow$): Suppose (λ^*, θ^*) is a solution to $LP_D(\mathcal{B})$ and $\lambda^* > 0$. Then since $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(\triangle Q\theta^*) - \lambda^*\mathbf{1} \geq -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q$ for all $i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}}$, we have that $\mathcal{AS}$ intersects the relative interior of $cone\left(G_{\bullet(\mathcal{B}\setminus\mathrm{Z}_{\mathcal{B}})}\right)$. Thus, since $dim(\mathcal{AS}) = k$, $dim\left(\mathcal{AS} \cap \mathcal{C}(\mathcal{B})\right) = \min\{k, h - |\mathrm{Z}_{\mathcal{B}}|\}$, which shows that $dim(\mathcal{IR}_{\mathcal{B}}) = k$ since $|\mathrm{Z}_{\mathcal{B}}| \leq h - k$.

($\Rightarrow$): We prove this direction using contradiction. Consider the following cases:

<u>Case 1</u>: $|\mathrm{Z}_{\mathcal{B}}| > h - k$

Notice that for each $i \in \mathrm{Z}_{\mathcal{B}}$, the column $G_{\bullet i}$ is unnecessary for the description of $\mathcal{AS} \cap \mathcal{C}(\mathcal{B})$. Therefore $(\mathcal{AS} \cap \mathcal{C}(\mathcal{B})) \subseteq cone\left(G_{\bullet(\mathcal{B}\setminus\mathrm{Z}_{\mathcal{B}})}\right)$. However, since $|\mathrm{Z}_{\mathcal{B}}| > h - k$, $dim\left(cone\left(G_{\bullet(\mathcal{B}\setminus\mathrm{Z}_{\mathcal{B}})}\right)\right) < k$ and thus $\mathcal{IR}_{\mathcal{B}}$ cannot be full dimensional, which is a contradiction.

Case 2: There does not exist a solution (λ^*, θ^*) to $LP_D(\mathcal{B})$ such that $\lambda^* > 0$.

If $LP_D(\mathcal{B})$ is infeasible, $\mathcal{IR}_{\mathcal{B}}$ is empty, which is a contradiction. On the other hand, suppose that for all solutions (λ^*, θ^*) to $LP_D(\mathcal{B})$, $\lambda^* \leq 0$. If this is the case then there does not exist $\theta \in S_\theta$ which lies in the relative interior of $\mathcal{IR}_{\mathcal{B}}$. Therefore $\mathcal{IR}_{\mathcal{B}}$ cannot be full dimensional, which is a contradiction.

Finding contradictions in both cases above shows that there must exist a solution (λ^*, θ^*) to $LP_D(\mathcal{B})$ such that $\lambda^* > 0$ and that $|\mathrm{Z}_{\mathcal{B}}| \leq h - k$. □

As a result of Proposition 4.34, $LP_D(\hat{\mathcal{B}})$ can be used to determine whether or not $\mathcal{IR}_{\hat{\mathcal{B}}}$ is full dimensional. We note here that due to the fact that S_θ is assumed to be a bounded polytope, LP_D will always have a bounded feasible region. Moreover, S_θ being bounded guarantees that every LP we introduce in this section will also have a bounded feasible region. Now, observe the initial tableaux for each of the examples:

Initial Tableau - Example 1

	w_1	w_2	w_3	z_1	z_2	z_3	
w_1	1	0	0	-1	0	0	$\theta_1 + 6$
w_2	0	1	0	0	-1	-1	$-\theta_2 - 1$
w_3	0	0	1	0	-1	-1	$\theta_1 - 3\theta_2 + 1$

Initial Tableau - Example 2

	w_1	w_2	w_3	z_1	z_2	z_3	
w_1	1	0	0	-5	-4	0	$\theta_1 + \theta_2 - 3$
w_2	0	1	0	-4	-5	0	$-\theta_1 + 3\theta_2 + 1$
w_3	0	0	1	1	0	-1	0

(4.26)

From these tableaux and (4.25) we obtain $LP_D(\mathcal{B}_0)$ for each example:

$$\begin{array}{ll} \max\limits_{\lambda,\theta} & \lambda \\ \text{s.t.} & \theta_1 - \lambda \geq -6 \\ & -\theta_2 - \lambda \geq 1 \\ & \theta_1 - 3\theta_2 - \lambda \geq -1 \\ & \theta \in [-2, 2]^2 \end{array}$$

$$\begin{array}{ll} \max\limits_{\lambda,\theta} & \lambda \\ \text{s.t.} & \theta_1 + \theta_2 - \lambda \geq 3 \\ & -\theta_1 + 3\theta_2 - \lambda \geq -1 \\ & \theta \in [-2, 2]^2 \end{array}$$

Notice from (4.22) that there are only two inequalities present in the LP for Example 2 because $w_3 \in \mathrm{Z}_{\mathcal{B}_0}$. The respective optimal solutions of each LP are $(\lambda^*, \theta_1^*, \theta_2^*) = (1, 0, -2)$ and $(\lambda^{**}, \theta_1^{**}, \theta_2^{**}) = (1, 2, 2)$. Hence, by Proposition 4.34, $\mathcal{IR}_{\mathcal{B}_0}$ is full dimensional for both examples. Thus, Proposition 4.34, together with the procedures of Section 4.4, allows for the completion of the intialization phase of Outline 4.33. The next task we address is that of step 2: "Determine the set of feasible bases which have invariancy regions that are: (i) adjacent to $\mathcal{IR}_{\mathcal{B}}$, and (ii) at least $(k-1)$–dimensional."

This can be done by finding the set of all facets of $\mathcal{IR}_{\mathcal{B}}$ and then, for each facet, determining all bases which have invariancy regions that are adjacent to $\mathcal{IR}_{\mathcal{B}}$ along that facet. Consider the following propositions, observations, and definitions.

Proposition 4.35. For a given complementary basis $\mathcal{B}$, if there exists $i \in \mathcal{B}$ and $\hat{\theta} \in S_\theta$ such that $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(q + \triangle Q\hat{\theta}) > 0$ for all $j \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right)$ and $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\hat{\theta}) = 0$, then $\hbar^i_{\mathcal{B}}$ is a facet of $\mathcal{IR}_{\mathcal{B}}$. Hence, if the following LP has a strictly positive optimal value, $\hbar^i_{\mathcal{B}}$ is a facet of $\mathcal{IR}_{\mathcal{B}}$.

$$\begin{aligned} LP_F(\mathcal{B}, i) := \quad & \max_{\lambda,\theta} \quad \lambda \\ & \text{s.t.} \quad \left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(\triangle Q\theta) - \lambda\mathbf{1} \geq -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} q \qquad \forall\, j \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right) \\ & \qquad \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(\triangle Q\theta) = -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q \\ & \qquad \theta \in S_\theta \end{aligned} \tag{4.27}$$

Proof. Suppose (λ^*, θ^*) is a solution to $LP_F(\mathcal{B}, i)$ and $\lambda^* > 0$. We first show that θ^* lies on the hyperplane $\hbar^i_{\mathcal{B}}$. Next we show that $\theta^* \in \mathcal{IR}_{\mathcal{B}}$ (4.3). Finally we show that any of the defining constraints of $\mathcal{IR}_{\mathcal{B}}$ which are formed by $\hbar^i_{\mathcal{B}}$ cannot be removed without adding new points to $\mathcal{IR}_{\mathcal{B}}$, i.e., $\hbar^i_{\mathcal{B}}$ forms a facet of $\mathcal{IR}_{\mathcal{B}}$.

From (4.23) notice that for each $\ell \in \mathrm{H}^i_{\mathcal{B}}$, $\hbar^i_{\mathcal{B}} = \hbar^\ell_{\mathcal{B}}$. So, since $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(\triangle Q\theta^*) = -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q$, it is clear that $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{\ell\bullet}(q + \triangle Q\theta^*) = 0$ for all $\ell \in \mathrm{H}^i_{\mathcal{B}}$. This also shows that θ^* lies on $\hbar^i_{\mathcal{B}}$. Next, recall from (4.22) that for all $j \in \mathrm{Z}_{\mathcal{B}}$, $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(q + \triangle Q\theta) = 0$ for all $\theta \in S_\theta$. Therefore for each $j \in \mathrm{Z}_{\mathcal{B}}$, the satisfaction of $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(q + \triangle Q\theta) \geq 0$ is trivial for all $\theta \in \mathcal{IR}_{\mathcal{B}}$. Thus, since θ^* satisfies the constraints of $LP_F(\mathcal{B}, i)$, $\theta^* \in \mathcal{IR}_{\mathcal{B}}$. Furthermore, since $\lambda^* > 0$, there must exist $\epsilon > 0$ such that all θ in the ball of radius ϵ centered at θ^* satisfy $\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(q + \triangle Q\theta) \geq 0$ for all $j \in \left(\mathcal{B} \setminus \left(\mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right)$. Hence, since θ^* lies on $\hbar^i_{\mathcal{B}}$, this ϵ-ball must contain a $\hat{\theta}$ satisfying all the defining inequalities of $\mathcal{IR}_{\mathcal{B}}$ except those which form $\hbar^i_{\mathcal{B}}$. Thus, the inqualities which form $\hbar^i_{\mathcal{B}}$ cannot be removed from the description of $\mathcal{IR}_{\mathcal{B}}$ without altering the polytope and therefore $\hbar^i_{\mathcal{B}}$ must be a facet. □

Observation 4.36. The converse of Proposition 4.35 also holds if $\mathcal{IR}_{\mathcal{B}}$ is full dimensional.

Observe $LP_F(\mathcal{B}_0, w_1)$ for each of the two examples:

$$\begin{aligned} \max_{\lambda,\theta} \quad & \lambda \\ \text{s.t.} \quad & \theta_1 = -6 \\ & -\theta_2 - \lambda \geq 1 \\ & \theta_1 - 3\theta_2 - \lambda \geq -1 \\ & \theta \in [-2, 2]^2 \end{aligned} \qquad\qquad \begin{aligned} \max_{\lambda,\theta} \quad & \lambda \\ \text{s.t.} \quad & \theta_1 + \theta_2 = 3 \\ & -\theta_1 + 3\theta_2 - \lambda \geq -1 \\ & \theta \in [-2, 2]^2 \end{aligned}$$

For Example 1, this LP is infeasible. For Example 2, the LP has an optimal solution of $(\lambda^{**}, \theta_1^{**}, \theta_2^{**}) = (6, 1, 2)$. Thus, by Proposition 4.35, $\hbar^{w_1}_{\mathcal{B}_0}$ is a facet of $\mathcal{IR}_{\mathcal{B}_0}$ for Example 2, but not for Example 1, where $\hbar^{w_1}_{\mathcal{B}_0} = \begin{cases} \{\theta \in S_\theta : \theta_1 = -6\} & \text{for Example 1} \\ \{\theta \in S_\theta : \theta_1 + \theta_2 = 3\} & \text{for Example 2} \end{cases}$. By following the same procedure for w_2 and w_3, we find that $\hbar^{w_2}_{\mathcal{B}_0}$ is the only facet of $\mathcal{IR}_{\mathcal{B}_0}$ for Example 1 and $\hbar^{w_1}_{\mathcal{B}_0}$ is the only facet of $\mathcal{IR}_{\mathcal{B}_0}$ for Example 2. Here $\hbar^{w_2}_{\mathcal{B}_0} = \begin{cases} \{\theta \in S_\theta : \theta_2 = -1\} & \text{for Example 1} \\ \{\theta \in S_\theta : -\theta_1 + 3\theta_2 = -1\} & \text{for Example 2} \end{cases}$. To see that the correct conclusions have been made about which hyperplanes form facets of $\mathcal{IR}_{\mathcal{B}_0}$, observe Figures 4.4 and 4.5.

We now continue our discussion and develop the tools necessary for determining bases having invariancy regions which are adjacent to a given invariancy region across a particular facet. Consider the following definition and subsequent proposition.

Definition 4.37. Given a complementary basis $\mathcal{B}$, the associated tableau $T_{\mathcal{B}}$, and distinct indices $i \in \mathcal{B}$ and $j \in \mathcal{E}$, a *pivot on* $(T_{\mathcal{B}})_{i,j}$ is the process of creating a new matrix T^* by performing elementary row operations on $T_{\mathcal{B}}$ so that $T^*_{i,j} = 1$ and $T^*_{\gamma,j} = 0$ for all $\gamma \in (\mathcal{B} \setminus \{i\})$.

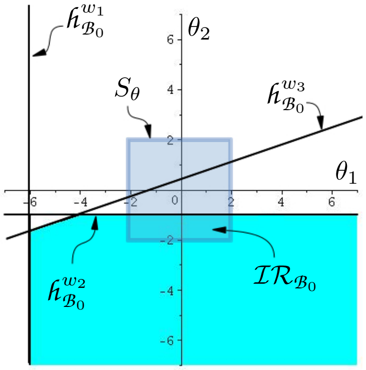


Figure 4.4: Example 1 - Only $\hbar^{w_2}_{\mathcal{B}_0}$ forms a facet of $\mathcal{IR}_{\mathcal{B}_0}$

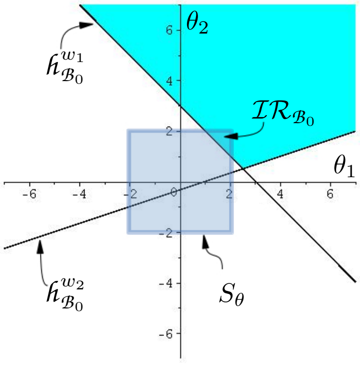


Figure 4.5: Example 2 - Only $\hbar^{w_1}_{\mathcal{B}_0}$ forms a facet of $\mathcal{IR}_{\mathcal{B}_0}$

Proposition 4.38. Let $\mathcal{B}$ be a feasible basis such that $\mathcal{IR}_{\mathcal{B}}$ is full dimensional and let $\hbar^i_{\mathcal{B}}$ be a facet of $\mathcal{IR}_{\mathcal{B}}$. For any complementary set $\mathcal{B}' \neq \mathcal{B}$ such that $|\mathcal{B} \cap \mathcal{B}'| \geq h - 2$, $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ if and only if one of the following conditions holds.

1. $(T_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$.

2. $(T_{\mathcal{B}})_{i,\bar{\imath}} = 0$, and there exists $j \in \mathcal{B}$ such that $(T_{\mathcal{B}})_{j,\bar{\imath}} > 0$, $(T_{\mathcal{B}})_{i,\bar{\jmath}} \neq 0$, and the following LP has

a strictly positive optimal value:

$$
\begin{aligned}
LP_A(\mathcal{B}, i, j) := & \\
& \max_{\lambda,\theta} \quad \lambda \\
& \text{s.t.} \quad \left(G^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet} (\triangle Q\theta) - \lambda\mathbf{1} \geq -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet} q \qquad \forall\, \gamma \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right) \\
& \qquad \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} (\triangle Q\theta) = -\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q \\
& \qquad \mathcal{Y}^{\xi}_{\mathcal{B}}(\theta) - \lambda \geq 0 \qquad \forall\, \xi \in \left(\mathcal{B} \setminus \left((\mathrm{Z}_{\mathcal{B}})' \cup (\mathrm{H}^i_{\mathcal{B}})' \cup \{i, j\}\right)\right) \\
& \qquad \theta \in S_\theta
\end{aligned}
\tag{4.28}
$$

where:

$$
\mathcal{Y}^{\xi}_{\mathcal{B}}(\theta) := \left(\left(G^{-1}_{\bullet\mathcal{B}}\right)_{\xi\bullet} - \frac{(T_{\mathcal{B}})_{\xi,\bar{\imath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}} \left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} - \left(\frac{(T_{\mathcal{B}})_{\xi,\bar{\jmath}}}{(T_{\mathcal{B}})_{i,\bar{\jmath}}} - \frac{(T_{\mathcal{B}})_{\xi,\bar{\imath}}(T_{\mathcal{B}})_{j,\bar{\jmath}}}{(T_{\mathcal{B}})_{i,\bar{\jmath}}(T_{\mathcal{B}})_{j,\bar{\imath}}}\right) \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\right) (q + \triangle Q\theta) \tag{4.29}
$$

$$
(\mathrm{Z}_{\mathcal{B}})' := \left\{\xi \in \mathcal{B} \setminus \{i, j\} : \mathcal{Y}^{\xi}_{\mathcal{B}}(\theta) = 0 \ \forall\, \theta \in S_\theta\right\} \tag{4.30}
$$

$$
(\mathrm{H}^i_{\mathcal{B}})' := \left\{\xi \in \mathcal{B} \setminus \{i, j\} : \exists \beta \neq 0 \text{ s.t. } \mathcal{Y}^{\xi}_{\mathcal{B}}(\theta) = \beta \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} (q + \triangle Q\theta) \ \forall\, \theta \in S_\theta\right\} \tag{4.31}
$$

(Note the similarity in notations between (4.22) and (4.30) as well as (4.23) and (4.31). The reasons for this similarity are shown in the following proof.)

Proof. (⇐): Consider the two conditions:

Condition 1: $(T_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$

By Observation 4.28, $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis. The adjacency of $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ along $\hbar^i_{\mathcal{B}}$ comes from the following facts:

(i) $\left(\mathcal{AS} \cap cone\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)\right)$ forms a facet of both $\mathcal{ID}_{\mathcal{B}}$ and $\mathcal{ID}_{\mathcal{B}'}$, and

(ii) $\mathcal{ID}_{\mathcal{B}}$ and $\mathcal{ID}_{\mathcal{B}'}$ share this facet if and only if $\hbar^i_{\mathcal{B}}$ is a facet of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$.

Condition 2: $(T_{\mathcal{B}})_{i,\bar{\imath}} = 0$, and there exists $j \in \mathcal{B}$ such that $(T_{\mathcal{B}})_{j,\bar{\imath}} > 0$, $(T_{\mathcal{B}})_{i,\bar{\jmath}} \neq 0$, and $LP_A(\mathcal{B}, i, j)$ has a nonpositive optimal value.

By Proposition 4.31, $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a basis. We will show that if there exists a solution (λ^*, θ^*) to $LP_A(\mathcal{B}, i, j)$ such that $\lambda^* > 0$, then $\hbar^i_{\mathcal{B}}$ forms a facet of $\mathcal{IR}_{\mathcal{B}'} = \left\{\theta \in S_\theta \subseteq \mathbb{R}^k : G^{-1}_{\bullet\mathcal{B}'}(q + \triangle Q\theta) \geq 0\right\}$. Now, since $(T_{\mathcal{B}})_{i,\bar{\imath}} = 0$, $(T_{\mathcal{B}})_{j,\bar{\imath}} > 0$ and $(T_{\mathcal{B}})_{i,\bar{\jmath}} \neq 0$, the tableau $T_{\mathcal{B}'}$ can be obtained from the tableau $T_{\mathcal{B}}$ in two steps: (i) create matrix T^* from $T_{\mathcal{B}}$ by performing a pivot on $(T_{\mathcal{B}})_{j,\bar{\imath}}$, and (ii) obtain $T_{\mathcal{B}'}$ from T^* by performing a pivot on $(T^*)_{i,\bar{\jmath}}$. Consider the following subset of $T_{\mathcal{B}}$:

	i	j	$\bar{\imath}$	$\bar{\jmath}$	
i	1	0	0	$(T_{\mathcal{B}})_{i,\bar{\jmath}}$	$\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\theta)$
γ	0	0	$(T_{\mathcal{B}})_{\gamma,\bar{\imath}}$	$(T_{\mathcal{B}})_{\gamma,\bar{\jmath}}$	$\left(G^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet}(q + \triangle Q\theta)$
j	0	1	$(T_{\mathcal{B}})_{j,\bar{\imath}}$	$(T_{\mathcal{B}})_{j,\bar{\jmath}}$	$\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(q + \triangle Q\theta)$

where γ represents any element of $\mathcal{B} \setminus \{i, j\}$. Then, by pivoting on $(T_{\mathcal{B}})_{j,\bar{\imath}}$, we obtain the following corresponding subset of T^*:

	i	j	$\bar{\imath}$	$\bar{\jmath}$	
i	1	0	0	$(T_{\mathcal{B}})_{i,\bar{\jmath}}$	$\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\theta)$
γ	0	$-\dfrac{(T_{\mathcal{B}})_{\gamma,\bar{\imath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}}$	0	$(T_{\mathcal{B}})_{\gamma,\bar{\jmath}} - (T_{\mathcal{B}})_{\gamma,\bar{\imath}}\dfrac{(T_{\mathcal{B}})_{j,\bar{\jmath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}}$	$\left(\left(G^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet} - \dfrac{(T_{\mathcal{B}})_{\gamma,\bar{\imath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}}\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}\right)(q + \triangle Q\theta)$
$\bar{\imath}$	0	$\dfrac{1}{(T_{\mathcal{B}})_{j,\bar{\imath}}}$	1	$\dfrac{(T_{\mathcal{B}})_{j,\bar{\jmath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}}$	$\dfrac{\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}(q + \triangle Q\theta)}{(T_{\mathcal{B}})_{j,\bar{\imath}}}$

Finally, by pivoting on $(T^*)_{i,\bar{\jmath}}$ we obtain the following corresponding subset of $T_{\mathcal{B}'}$ (for the sake of space, we only show the RHS):

$\bar{\jmath}$	$\dfrac{\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\theta)}{(T_{\mathcal{B}})_{i,\bar{\jmath}}}$
γ	$\mathcal{Y}^{\gamma}_{\mathcal{B}}(\theta)$
$\bar{\imath}$	$\dfrac{1}{(T_{\mathcal{B}})_{j,\bar{\imath}}}\left(\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} - \dfrac{(T_{\mathcal{B}})_{j,\bar{\jmath}}}{(T_{\mathcal{B}})_{i,\bar{\jmath}}}\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\right)(q + \triangle Q\theta)$

This subset of tableau $T_{\mathcal{B}'}$ provides us with the following information:

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\bar{\jmath}\bullet}(q + \triangle Q\theta) = \frac{\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\theta)}{(T_{\mathcal{B}})_{i,\bar{\jmath}}} \quad \text{for all } \theta \in S_\theta, \tag{4.32}$$

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\bar{\imath}\bullet}(q + \triangle Q\theta) = \frac{1}{(T_{\mathcal{B}})_{j,\bar{\imath}}}\left(\left(G^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} - \frac{(T_{\mathcal{B}})_{j,\bar{\jmath}}}{(T_{\mathcal{B}})_{i,\bar{\jmath}}}\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\right)(q + \triangle Q\theta) \quad \text{for all } \theta \in S_\theta, \tag{4.33}$$

and

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\gamma\bullet}(q + \triangle Q\theta) = \mathcal{Y}^{\gamma}_{\mathcal{B}}(\theta) \text{ for all } \gamma \in \mathcal{B}' \setminus \{\bar{\imath}, \bar{\jmath}\} \text{ and } \theta \in S_\theta. \tag{4.34}$$

From (4.32) we have $\hbar^{\bar{\jmath}}_{\mathcal{B}'} = \hbar^{i}_{\mathcal{B}}$. Therefore, by Proposition 4.35, if the following conditions hold,

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\gamma\bullet}(q + \triangle Q\theta^*) > 0 \text{ for all } \gamma \in \left(\mathcal{B}' \setminus \left(\mathrm{Z}_{\mathcal{B}'} \cup \mathrm{H}^{\bar{\jmath}}_{\mathcal{B}'} \cup \{\bar{\jmath}\}\right)\right) \tag{4.35}$$

and

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\bar{\jmath}\bullet}(q + \triangle Q\theta^*) = 0, \tag{4.36}$$

then $\hbar^{i}_{\mathcal{B}}$ is a facet of $\mathcal{IR}_{\mathcal{B}'}$.

Now, since (λ^*, θ^*) satisfies (4.28) and $\lambda^* > 0$, we have:

$$\left(G^{-1}_{\bullet\mathscr{B}}\right)_{\gamma\bullet}(q + \triangle Q\theta^*) > 0 \text{ for all } \gamma \in \left(\mathscr{B} \setminus \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{H}^{i}_{\mathscr{B}} \cup \{i\}\right)\right), \tag{4.37}$$

$$\left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet}(q + \triangle Q\theta^*) = 0, \tag{4.38}$$

and

$$\mathscr{Y}^{\xi}_{\mathscr{B}}(\theta) > 0 \text{ for all } \xi \in \left(\mathscr{B} \setminus \left((\mathrm{Z}_{\mathscr{B}})' \cup (\mathrm{H}^{i}_{\mathscr{B}})' \cup \{i, j\}\right)\right). \tag{4.39}$$

From (4.32) we observe that (4.36) is given by (4.38), so we focus on (4.35). To show that (4.35) is satisfied, we need to show: (i) if $\overline{\imath} \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^{\overline{\jmath}}_{\mathscr{B}'}\right)\right)$ then $\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\overline{\imath}\bullet}(q + \triangle Q\theta^*) > 0$, and (ii) for all $\gamma \in \mathscr{B}' \setminus \{\overline{\imath}, \overline{\jmath}\}$, if $\gamma \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^{\overline{\jmath}}_{\mathscr{B}'}\right)\right)$ then $\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\gamma\bullet}(q + \triangle Q\theta^*) > 0$.

Recall that $j \neq i$ and and notice the following:

- If $j \in \left(\mathscr{B} \setminus \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{H}^{i}_{\mathscr{B}}\right)\right)$, then (4.33), (4.37) and (4.38) imply $\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\overline{\imath}\bullet}(q + \triangle Q\theta^*) > 0$ since $(T_{\mathscr{B}})_{j,\overline{\imath}} > 0$.
- If $j \in \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{H}^{i}_{\mathscr{B}}\right)$, then from (4.32), (4.33) and the definitions of $\mathrm{Z}_{\mathscr{B}}$ and $\mathrm{H}^{i}_{\mathscr{B}}$ ((4.22) and (4.23)), we have $\overline{\imath} \in \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^{\overline{\jmath}}_{\mathscr{B}'}\right)$.

Thus, since j must be a member of either $\left(\mathscr{B} \setminus \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{H}^{i}_{\mathscr{B}}\right)\right)$ or $\left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{H}^{i}_{\mathscr{B}}\right)$, it is clear that if $\overline{\imath} \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^{\overline{\jmath}}_{\mathscr{B}'}\right)\right)$, then $\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\overline{\imath}\bullet}(q + \triangle Q\theta^*) > 0$.

Now consider $\gamma \in \mathscr{B}' \setminus \{\overline{\imath}, \overline{\jmath}\}$. Notice that from (4.32), (4.34) and the definitions of $\mathrm{Z}_{\mathscr{B}}$, $(\mathrm{Z}_{\mathscr{B}})'$, $\mathrm{H}^{i}_{\mathscr{B}}$, and $(\mathrm{H}^{i}_{\mathscr{B}})'$ ((4.22),(4.23),(4.30) and (4.31)) we have:

$$\gamma \in (\mathrm{Z}_{\mathscr{B}})' \Leftrightarrow \gamma \in \mathrm{Z}_{\mathscr{B}'} \tag{4.40}$$

and

$$\gamma \in (\mathrm{H}^{i}_{\mathscr{B}})' \Leftrightarrow \gamma \in \mathrm{H}^{\overline{\jmath}}_{\mathscr{B}'} \tag{4.41}$$

From these facts and (4.39) we conclude that if $\gamma \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^{\overline{\jmath}}_{\mathscr{B}'} \cup \{\overline{\imath}, \overline{\jmath}\}\right)\right)$, then $\gamma \in \left(\mathscr{B} \setminus \left((\mathrm{Z}_{\mathscr{B}})' \cup (\mathrm{H}^{i}_{\mathscr{B}})' \cup \{i, j\}\right)\right)$ and thus $\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\gamma\bullet}(q + \triangle Q\theta^*) > 0$.

We have now shown that both (4.35) and (4.36) are satisfied and thus $\mathscr{h}^{i}_{\mathscr{B}}$ forms a facet of $\mathcal{IR}_{\mathscr{B}'}$.

($\Rightarrow$): Since $\mathcal{B}'$ is a complementary set such that $\mathcal{B}' \neq \mathcal{B}$, $|\mathcal{B} \cap \mathcal{B}'| \geq h-2$, and $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$, we must have one of the following two cases:

Case 1: $|\mathcal{B} \cap \mathcal{B}'| = h-1$

In this case $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ if and only if $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis. Additionally, $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis if and only if $(T_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$. Hence, in this case Condition 1 is satisfied.

Case 2: $|\mathcal{B} \cap \mathcal{B}'| = h-2$

In this case, the fact that $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ implies that there exists $j \in \mathcal{B}$ such that $\mathcal{B}' = (\mathcal{B} \setminus \{i,j\}) \cup \{\bar{\imath},\bar{\jmath}\}$ is a basis and $\mathcal{C}(\mathcal{B})$ is adjacent to $\mathcal{C}(\mathcal{B}')$ along the facet $cone\left(G^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Then by Proposition 4.32, $(T_{\mathcal{B}})_{i,\bar{\imath}} = 0$, $(T_{\mathcal{B}})_{j,\bar{\imath}} > 0$, and $(T_{\mathcal{B}})_{i,\bar{\jmath}} \neq 0$. Furthermore, the work done in the proof of Proposition 4.32 shows that $\mathcal{C}(\mathcal{B})$ and $\mathcal{C}(\mathcal{B}')$ are adjacent along $cone\left(G^{-1}_{\bullet(\mathcal{B}'\setminus\{\bar{\jmath}\})}\right)$, i.e., $\hbar^i_{\mathcal{B}} = \hbar^{\bar{\jmath}}_{\mathcal{B}'}$. Therefore, the fact that $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ also implies that there exists $\theta^* \in \mathcal{IR}_{\mathcal{B}}$ which is also in $\mathcal{IR}_{\mathcal{B}'}$ and lies in the relative interiors of both the facet of $\mathcal{IR}_{\mathcal{B}}$ formed by $\hbar^i_{\mathcal{B}}$ and the facet of $\mathcal{IR}_{\mathcal{B}'}$ formed by $\hbar^{\bar{\jmath}}_{\mathcal{B}'}$. Recognize that this can only be true if all of the following hold:

$$\left(G^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet}(q + \triangle Q\theta^*) > 0 \text{ for all } \gamma \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right), \tag{4.42}$$

$$\left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}(q + \triangle Q\theta^*) = 0, \tag{4.43}$$

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\gamma\bullet}(q + \triangle Q\theta^*) > 0 \text{ for all } \gamma \in \left(\mathcal{B}' \setminus \left(\mathrm{Z}_{\mathcal{B}'} \cup \mathrm{H}^{\bar{\jmath}}_{\mathcal{B}'} \cup \{\bar{\jmath}\}\right)\right), \tag{4.44}$$

and

$$\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\bar{\jmath}\bullet}(q + \triangle Q\theta^*) = 0. \tag{4.45}$$

Furthermore, notice that from (4.32), (4.40) and (4.41), the four conditions above imply that there must exist $\lambda^* > 0$ such that (λ^*, θ^*) is a solution to (4.28). Hence, in this case Condition 2 is satisfied. □

Consider again the two examples. Recall from (4.26) that for Example 1, $(T_{\mathcal{B}_0})_{w_2,z_2} = -1$ and for Example 2, $(T_{\mathcal{B}_0})_{w_1,z_1} = 1$. By condition 1 of Proposition 4.38, performing pivots on these elements of the respective tableaux provides new bases $\mathscr{B}_1^{ex1} = \{w_1, z_2, w_3\}$ and $\mathscr{B}_1^{ex2} = \{z_1, w_2, w_3\}$ which have invariancy regions which are adjacent to $\mathcal{IR}_{\mathscr{B}_0}$ for their respective examples.

At this point let us discuss each example separately in order to highlight certain key qualities present in each. We begin with Example 1. Using $LP_F(\mathscr{B}_1^{ex1}, w_1)$, $LP_F(\mathscr{B}_1^{ex1}, z_2)$ and $LP_F(\mathscr{B}_1^{ex1}, w_3)$ we can determine that $\hbar^{z_2}_{\mathscr{B}_1^{ex1}} = \{\theta \in S_\theta : \theta_2 = -1\}$ and $\hbar^{w_3}_{\mathscr{B}_1^{ex1}} = \{\theta \in S_\theta : \theta_1 - 2\theta_2 = -2\}$ form the facets of $\mathcal{IR}_{\mathscr{B}_1^{ex1}}$. Since we can conclude from Corollary 4.30 that $\mathcal{IR}_{\mathscr{B}_0}$ is the only invariancy region adjacent to $\mathcal{IR}_{\mathscr{B}_1^{ex1}}$ across $\hbar^{z_2}_{\mathscr{B}_1^{ex1}}$, we need only look for adjacent regions across $\hbar^{w_3}_{\mathscr{B}_1^{ex1}}$. Observe $T_{\mathscr{B}_1^{ex1}}$:

	w_1	w_2	w_3	z_1	z_2	z_3	
w_1	1	0	0	-1	0	0	$\theta_1 + 6$
z_2	0	-1	0	0	1	1	$\theta_2 + 1$
w_3	0	-1	1	0	0	0	$\theta_1 - 2\theta_2 + 2$

(4.46)

Here $(T_{\mathscr{B}_1^{ex1}})_{w_3,z_3} = 0$ and thus condition 1 of Proposition 4.38 does not apply and we must use condition 2. Recognize that $(T_{\mathscr{B}_1^{ex1}})_{z_2,z_3} = 1$ is the only positive element of the column of $T_{\mathscr{B}_1^{ex1}}$ associated with z_3. Since $(T_{\mathscr{B}_1^{ex1}})_{w_3,w_2} = -1$, condition 2 is satisfied if the optimal solution to $LP_A(\mathscr{B}_1^{ex1}, w_3, z_2)$ is strictly positive. Using (4.28), (4.29), (4.30) and (4.31) we obtain $LP_A(\mathscr{B}_1^{ex1}, w_3, z_2)$:

$$\begin{array}{rc} \max\limits_{\lambda,\theta} & \lambda \\ \text{s.t.} & \theta_1 - \lambda \geq -6 \\ & \theta_2 - \lambda \geq -1 \\ & \theta_1 - 2\theta_2 = -2 \\ & \theta \in [-2, 2]^2 \end{array}$$

The optimal solution to this LP is $(\lambda^*, \theta_1^*, \theta_2^*) = (3, 2, 2)$, which shows that condition 2 is satisfied and consequently that the invariancy region associated with basis $\mathscr{B}_2^{ex1} = \{w_1, w_2, z_3\}$ is adjacent to $\mathcal{IR}_{\mathscr{B}_1^{ex1}}$ across $\hbar^{w_3}_{\mathscr{B}_1^{ex1}}$. Using $LP_F(\mathscr{B}_2^{ex1}, w_1)$, $LP_F(\mathscr{B}_2^{ex1}, w_2)$ and $LP_F(\mathscr{B}_2^{ex1}, z_3)$ we find that $\hbar^{z_3}_{\mathscr{B}_2^{ex1}} = \{\theta \in S_\theta : -\theta_1 + 2\theta_2 = 2\}$ forms the only facet of $\mathcal{IR}_{\mathscr{B}_2^{ex1}}$. From Corollary 4.30 we know that $\mathcal{IR}_{\mathscr{B}_1^{ex1}}$ is the only invariancy region adjacent to $\mathcal{IR}_{\mathscr{B}_2^{ex1}}$ across this facet. Thus, we have completed the partition of S_θ for Example 1.

Let us now return our focus to Example 2. Observe $T_{\mathcal{B}_1^{ex2}}$:

	w_1	w_2	w_3	z_1	z_2	z_3	
z_1	-$\frac{1}{5}$	0	0	1	$\frac{4}{5}$	0	$-\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 + \frac{3}{5}$
w_2	-$\frac{4}{5}$	1	0	0	-$\frac{9}{5}$	0	$-\frac{9}{5}\theta_1 + \frac{11}{5}3\theta_2 + \frac{17}{5}$
w_3	$\frac{1}{5}$	0	1	0	-$\frac{4}{5}$	-1	$\frac{1}{5}\theta_1 + \frac{1}{5}\theta_2 - \frac{3}{5}$

(4.47)

Using $LP_F(\mathcal{B}_1^{ex2}, z_1)$, $LP_F(\mathcal{B}_1^{ex2}, w_2)$ and $LP_F(\mathcal{B}_1^{ex2}, w_3)$ we can determine that $\hbar^{z_1}_{\mathcal{B}_1^{ex2}} = \left\{\theta \in S_\theta : -\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 = -\frac{3}{5}\right\}$ and $\hbar^{w_3}_{\mathcal{B}_1^{ex2}} = \{\theta \in S_\theta : \frac{1}{5}\theta_1 + \frac{1}{5}\theta_2 = \frac{3}{5}\}$ form the facets of $\mathcal{IR}_{\mathcal{B}_1^{ex2}}$. However, from (4.23) recognize that $w_3 \in \mathrm{H}^{z_1}_{\mathcal{B}_1^{ex2}}$ and thus $\hbar^{z_1}_{\mathcal{B}_1^{ex2}} = \hbar^{w_3}_{\mathcal{B}_1^{ex2}}$. Furthermore, since the RHS of $T_{\mathcal{B}_1^{ex2}}$ associated with z_1 is a negative multiple of the RHS of $T_{\mathcal{B}_1^{ex2}}$ associated with w_3, the defining inequalities of $\mathcal{IR}_{\mathcal{B}_1^{ex2}}$ associated with these variables define opposite half-spaces. Hence, $\mathcal{IR}_{\mathcal{B}_1^{ex2}}$ cannot be full dimensional. The following Corollaries provide the theory necessary for determining when a given invariancy region is not full dimensional and ensuring that only invariancy regions of dimension at least $k-1$ are discovered.

Corollary 4.39. Let distinct feasible complementary bases $\mathcal{B}'$ and $\mathcal{B}$ be given which satisfy: (i) $dim(\mathcal{IR}_{\mathcal{B}}) = k$, (ii) $|\mathcal{B}' \cap \mathcal{B}| \geq h-2$, and (iii) $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\hbar^i_{\mathcal{B}}$. Then $dim\left(\mathcal{IR}_{\mathcal{B}'}\right) \geq k-1$. Furthermore, $dim\left(\mathcal{IR}_{\mathcal{B}'}\right) = k-1$ if and only if the set

$$\mathrm{L}^i_{\mathcal{B},\mathcal{B}'} := \left\{\ell \in B' : \exists\, \beta_\ell > 0 \text{ s.t. } \left(\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\ell\bullet} - \beta_\ell \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\right)(q + \triangle Q\theta) = 0\right\} \tag{4.48}$$

is nonempty.

Proof. First notice that $dim\left(\mathcal{IR}_{\mathcal{B}'}\right) \geq k-1$ since $\hbar^i_{\mathcal{B}}$ forms a $k-1$ dimensional facet of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$, as shown in the proof of Proposition 4.38. We now prove that $dim\left(\mathcal{IR}_{\mathcal{B}'}\right) = k-1$ if and only if there exists $\ell \in \mathcal{B}'$ and $\beta_\ell > 0$ such that $\left(\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\ell\bullet} - \beta_\ell \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\right)(q + \triangle Q\theta) = 0$.

($\Rightarrow$): $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ share a $k-1$ dimensional facet, which is formed by $\hbar^i_{\mathcal{B}}$. Notice that there cannot exist $\gamma \in \mathcal{B}'$ such that $\hbar^\gamma_{\mathcal{B}}$ intersects the relative interior of the facet of $\mathcal{IR}_{\mathcal{B}'}$ formed by $\hbar^i_{\mathcal{B}}$ unless $\hbar^\gamma_{\mathcal{B}} = \hbar^i_{\mathcal{B}}$. Therefore, since $dim\left(\mathcal{IR}_{\mathcal{B}'}\right) \geq k-1$, there must exist $\ell \in \mathcal{B}'$ such that $\hbar^\ell_{\mathcal{B}} = \hbar^i_{\mathcal{B}}$ and the half-space $\left(\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\ell\bullet}\right)(q + \triangle Q\theta) \geq 0$ contains $\mathcal{IR}_{\mathcal{B}}$, i.e., there must exist $\beta_\ell > 0$ such that $\left(\left(G^{-1}_{\bullet\mathcal{B}'}\right)_{\ell\bullet} - \beta_\ell \left(G^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\right)(q + \triangle Q\theta) = 0$.

($\Leftarrow$): Since there exists $\ell \in \mathscr{B}'$ and $\beta_\ell > 0$ such that $\left(\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\ell\bullet} - \beta_\ell \left(G^{-1}_{\bullet\mathscr{B}}\right)_{i\bullet}\right)(q + \triangle Q\theta) = 0$, the half-space $\left(\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\ell\bullet}\right)(q + \triangle Q\theta) \geq 0$ contains $\mathcal{IR}_{\mathscr{B}}$. Thus, since complementary cones are disjoint in their relative interiors, and since $\mathcal{IR}_{\mathscr{B}}$ and $\mathcal{IR}_{\mathscr{B}'}$ share a $k-1$ dimensional facet, $\mathcal{IR}_{\mathscr{B}'}$ must be $k-1$ dimensional. □

Proposition 4.38 provides a strategy for determining the invariancy regions which are adjacent to a given full dimensional invariancy region. Corollary 4.39 provides a strategy for determining when an invariancy region is $k-1$ dimensional. Now, given a $k-1$ dimensional invariancy region we need to be able to determine adjacent invariancy regions which have dimension at least $k-1$. For this purpose we introduce the following corollary which is quite similar to Proposition 4.38.

Corollary 4.40. Suppose that $\mathscr{B}$ and $\mathscr{B}'$ are two complementary bases such that $|\mathscr{B} \cap \mathscr{B}'| \geq h-2$, $\mathcal{IR}_{\mathscr{B}}$ and $\mathcal{IR}_{\mathscr{B}'}$ are adjacent along $\hbar^i_{\mathscr{B}}$, and $dim\left(\mathcal{IR}_{\mathscr{B}'}\right) = k-1$. For a complementary basis $\mathscr{B}''$ such that $|\mathscr{B}' \cap \mathscr{B}''| \geq h-2$, $\mathcal{IR}_{\mathscr{B}'}$ and $\mathcal{IR}_{\mathscr{B}''}$ are adjacent along $\hbar^i_{\mathscr{B}}$ if and only if one of the following conditions holds.

1. There exists $\ell \in \mathrm{L}^i_{\mathscr{B},\mathscr{B}'} \cup \left(\bigcup_{j \in \mathrm{L}^i_{\mathscr{B},\mathscr{B}'}} \mathrm{H}^j_{\mathscr{B}'}\right)$ such that $\left(T_{\mathscr{B}'}\right)_{\ell,\overline{\ell}} = 0$.
2. There exists $\ell \in \mathcal{L}^i_{\mathscr{B},\mathscr{B}'} \cup \left(\bigcup_{j \in \mathcal{L}^i_{\mathscr{B},\mathscr{B}'}} \mathrm{H}^j_{\mathscr{B}'}\right)$ such that $\left(T_{\mathscr{B}'}\right)_{\ell,\overline{\ell}} \neq 0$ and there exists $j \in \mathscr{B}' \setminus \{\ell\}$ such that $\left(T_{\mathscr{B}}\right)_{j,\overline{\ell}} > 0$, $\left(T_{\mathscr{B}}\right)_{\ell,\overline{j}} \neq 0$, and the following LP has a strictly positive optimal value:

$$LP_{A2}(\mathscr{B}, \ell, j) :=$$

$$\begin{array}{lll} \max\limits_{\lambda,\theta} & \lambda & \\ \text{s.t.} & \left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\gamma\bullet}(\triangle Q\theta) - \lambda\mathbf{1} \geq -\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\gamma\bullet} q & \forall\, \gamma \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^\ell_{\mathscr{B}'} \cup \{\ell\}\right)\right) \\ & \left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\ell\bullet}(\triangle Q\theta) = -\left(G^{-1}_{\bullet\mathscr{B}'}\right)_{\ell\bullet} q & \\ & \mathscr{Y}^\xi_{\mathscr{B}'}(\theta) - \lambda \geq 0 & \forall\, \xi \in \left(\mathscr{B}' \setminus \left((\mathrm{Z}_{\mathscr{B}})'' \cup (\mathrm{H}^i_{\mathscr{B}})'' \cup \{\ell, j\}\right)\right) \\ & \theta \in S_\theta & \end{array} \tag{4.49}$$

Proof. Notice that for all $j \in \mathcal{L}^i_{\mathscr{B},\mathscr{B}'}$ we have $\hbar^j_{\mathscr{B}'} = \hbar^i_{\mathscr{B}}$. Thus, from (4.23) it is clear that $\hbar^\ell_{\mathscr{B}'} = \hbar^i_{\mathscr{B}}$ if and only if $\ell \in \mathcal{L}^i_{\mathscr{B},\mathscr{B}'} \cup \left(\bigcup_{j \in \mathcal{L}^i_{\mathscr{B},\mathscr{B}'}} \mathrm{H}^j_{\mathscr{B}'}\right)$. The remainder of the proof is analogous to the proof of Proposition 4.38. □

Observation 4.41. As a result of Corollary 4.40, the statement of Corollary 4.39 can be relaxed to the following: Let distinct feasible complementary bases $\mathscr{B}'$ and $\mathscr{B}$ be given which satisfy: (i)

$dim(\mathcal{IR}_{\mathcal{B}}) \geq k-1$, (ii) $|\mathcal{B}' \cap \mathcal{B}| \geq h-2$, and (iii) $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\hbar^i_{\mathcal{B}}$. Then $dim\,(\mathcal{IR}_{\mathcal{B}'}) \geq k-1$. Furthermore, $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$ if and only if $\mathrm{L}^i_{\mathcal{B},\mathcal{B}'}$ is nonempty.

We return to our consideration of Example 2. Observe from (4.26), (4.47) and (4.48) that $w_3 \in \mathrm{L}^{w_1}_{\mathcal{B}_0,\,\mathcal{B}^{ex2}_1}$. Hence, by Corollary 4.39 we know that $dim\left(\mathcal{IR}_{\mathcal{B}^{ex2}_1}\right) = k-1 = 1$ and therefore we need to use the results of Corollary 4.40 to find other invariancy regions adjacent to $\mathcal{IR}_{\mathcal{B}^{ex2}_1}$ (as well as $\mathcal{IR}_{\mathcal{B}_0}$). Recognize from (4.47) that $\left(T_{\mathcal{B}^{ex2}_1}\right)_{w_3,z_3} = -1$. Thus, condition 1 of Corollary 4.40 is satisfied. By pivoting on this element we obtain a new basis $\mathcal{B}^{ex2}_2 = \{z_1, w_2, z_3\}$. It is easy to generate $T_{\mathcal{B}^{ex2}_1}$ and verify that $\mathcal{IR}_{\mathcal{B}^{ex2}_2}$ is full dimensional. We omit the rest of the details of Example 2 as they are analogous to those already presented. We return to these examples in Section 4.4 where we discuss determining an initial feasible basis.

Recall that replacing a single element of a complementary basis with its complement is a diagonal pivot, while replacing two elements with their complements is an exchange pivot. Thus, Proposition 4.38 and Corollary 4.40 provide conditions under which diagonal and exchange pivots from a given complementary basis $\mathcal{B}$ will yield new complementary bases whose associated invariancy regions are: (i) adjacent to $\mathcal{IR}_{\mathcal{B}}$, and (ii) at least $(k-1)$–dimensional.

Proposition 4.42. Let distinct bases $\mathcal{B}_i$ and $\mathcal{B}_j$ for which $\mathcal{IR}_{\mathcal{B}_i}$ and $\mathcal{IR}_{\mathcal{B}_j}$ are both full dimensional be given. Then $\mathcal{IR}_{\mathcal{B}_i}$ is adjacent to $\mathcal{IR}_{\mathcal{B}_j}$ if and only if $\mathcal{B}_j$ can be obtained from $\mathcal{B}_i$ by a sequence of pivots as described in Proposition 4.38 and Corollary 4.40.

Proof. As the reverse direction of the proposition is clear, we prove the forward direction. We will prove the contrapositive. By Proposition 4.25, for any basis $\mathcal{B}$, all bases adjacent to $\mathcal{B}$ whose complementary cones have at least a $k-1$ dimensional intersection with $\mathcal{AS}$ can be found by a pivot as described in either Proposition 4.38 or Corollary 4.40. Therefore, if there does not exist a sequence of these pivots by which $\mathcal{B}_j$ can be obtained from $\mathcal{B}_i$, then there cannot exist a sequence of invariancy regions $\{\mathcal{IR}_{\mathcal{B}_\ell}\}_{\ell=i+1}^{j-1}$ such that $\mathcal{C}(\mathcal{B}_\gamma)$ and $\mathcal{C}(\mathcal{B}_{\gamma+1})$ are adjacent for all $\gamma \in \{i, \ldots, j-1\}$ and $dim\,(\mathcal{C}(\mathcal{B}_\gamma) \cap \mathcal{AS}) = k-1$ for all $\gamma \in \{i+1, \ldots, j-1\}$. Thus, by Theorem 5.10 of [18], $\mathcal{IR}_{\mathcal{B}_i}$ cannot be adjacent to $\mathcal{IR}_{\mathcal{B}_j}$. □

At this point we have developed the theory necessary for partitioning S_θ, given an initial basis $\mathcal{B}_0$ such that $dim\,(\mathcal{IR}_{\mathcal{B}_0}) \geq k-1$. The algorithm is presented as Algorithm 4.1 in Section 4.5.

4.4 Phase 1: Determining an initial feasible solution

In this section we develop a method for determining an initial feasible solution to the mpLCP (4.1) which provides a starting point when partitioning the parameter space S_θ. Thus, we seek a basis $\mathscr{B}_0$ such that $dim\left(\mathcal{IR}_{\mathscr{B}_0}\right) \geq k-1$. We present the algorithm for finding $\mathscr{B}_0$ in Section 4.5.

We now discuss the techniques we use to obtain an initial basis $\mathscr{B}_0$ such that $dim(\mathcal{IR}_{\mathscr{B}_0}) \geq k-1$. We assume throughout this discussion that $\mathbf{0} \in S_\theta$. Recognize that this assumption is not restrictive because it can be achieved by a simple translation when necessary. Define the augmented phase 1 multiparametric LCP, mpLCP$_{ph1}$:

$$\begin{aligned} w - Mz &= q + \triangle Q\theta + r\phi \\ w^\top z &= 0 \\ w, z &\geq 0 \end{aligned} \tag{4.50}$$

where $\phi \in \mathbb{R}$ is an additional parameter and $r \in \mathbb{R}^h$ is defined so that

$$r_i = \begin{cases} |q_i| + 1 & \text{if } q_i \leq 0 \\ 0 & \text{otherwise} \end{cases} \qquad \text{for each } i \in \{1, \ldots, h\}. \tag{4.51}$$

Notice that mpLPC$_{ph1}$ (4.50) is a variant of mpLCP (4.1) in which k is replaced by $k+1$. Therefore, all definitions and theory presented for mpLCP directly apply to mpLPC$_{ph1}$. Throughout the following discussion we will use the notation $\mathcal{IR}_{\mathscr{B}}^{ph1}$ and $(\hbar_{\mathscr{B}}^{i})^{ph1}$ to denote the respective analogues of $\mathcal{IR}_{\mathscr{B}}$ and $\hbar_{\mathscr{B}}^{i}$ for mpLPC$_{ph1}$.

Recognize that we impose no lower or upper bound on the value of ϕ. As a result, the phase 1 counterpart to any LP presented for phase 2 can yield an unbounded solution. We note that this is not a problem since an unbounded maximization (minimization) problem still yields a strictly positive (negative) optimal value, which is the condition that must be verified for the majority of the LPs we have introduced.

Proposition 4.43. The complementary basis $\mathscr{B}^* := \{1, \ldots, h\}$ is feasible to mpLCP$_{ph1}$ (4.50) and $\mathcal{IR}_{\mathscr{B}^*}^{ph1}$ is full dimensional.

Proof. Since $G_{\bullet\mathscr{B}^*} = I$, $\mathcal{IR}_{\mathscr{B}^*}^{ph1} = \{(\theta, \phi) \in S_\theta \times \mathbb{R} : q + \triangle Q\theta + r\phi \geq 0\}$. Observe from this and (4.51) that $(\theta, \phi) = (\mathbf{0}, 1) \in \mathcal{IR}_{\mathscr{B}^*}^{ph1}$. Thus $\mathcal{IR}_{\mathscr{B}^*}^{ph1} \neq \emptyset$ and therefore $\mathscr{B}^*$ is feasible to mpLCP$_{ph1}$. Furthermore, since the system of inequalities $q + \triangle Q\theta + r\phi \geq 0$ is satisfied strictly at $(\theta, \phi) = (\mathbf{0}, 1)$, there must exist $\epsilon > 0$ such that $B_\epsilon\left((\mathbf{0}, 1)\right) \subset \mathcal{IR}_{\mathscr{B}^*}^{ph1}$. Hence, $\mathcal{IR}_{\mathscr{B}^*}^{ph1}$ is full dimensional. □

Proposition 4.43 shows that a full dimensional invariancy region for mpLCP$_{ph1}$ is immediately available. Thus, a very simple strategy for determining an initial basis $\mathcal{B}_0$ is to determine the facets of $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$, determine the bases whose phase 1 invariancy regions are adjacent to $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$ across each facet, and then repeat this procedure for each newly discovered invariancy region. Each time a new basis $\mathcal{B}$ is discovered, $LP_D(\mathcal{B})$ (4.25) can be solved to determine whether or not $\mathcal{IR}_{\mathcal{B}}$ is full dimensional. We then continue partitioning $S_\theta \times \mathbb{R}$ in the same way that we discussed partitioning S_θ in Section 3, and stop once a basis with a full dimensional invariancy region is discovered. If no such basis is discovered throughout the procedure, we can conclude that no such basis exists. Note that if no basis exists which has a full dimensional invariancy region, then there is no need to search for bases whose invariancy regions are $(k-1)$–dimensional. Although this procedure is a brute force method, it serves as a good foundation for the procedure we will ultimately use.

Recognize that for any phase 1 invariancy region $\mathcal{IR}^{ph1}_{\mathcal{B}}$, the phase 2 invariancy region $\mathcal{IR}_{\mathcal{B}}$ is precisely the intersection of $\mathcal{IR}^{ph1}_{\mathcal{B}}$ with the hyperplane $\phi = 0$. Thus, in order to improve the technique discussed above, we would like to determine the facets of an invariancy region $\mathcal{IR}^{ph1}_{\mathcal{B}}$ across which we are most likely to find an adjacent invariancy region $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ such that the intersection of $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ with the hyperplane $\phi = 0$ has a dimension of at least $k-1$. With this in mind, consider the following LP:

$$LP_S(\mathcal{B}) := \begin{array}{rc} \min\limits_{\theta,\phi} & \phi \\ \text{s.t.} & G^{-1}_{\bullet\mathcal{B}}(\triangle Q\theta + r\phi) \geq -G^{-1}_{\bullet\mathcal{B}}q \\ & \theta \in S_\theta \end{array} \tag{4.52}$$

This LP gives rise to the following proposition.

Proposition 4.44. If M is a Q_0 matrix, then the mpLCP (4.1) is feasible if and only if there exists a complementary basis $\mathcal{B}$ for which $LP_S(\mathcal{B})$ (4.52) has a nonpositive optimal value.

Proof. ($\Rightarrow$): If mpLCP (4.1) is feasible then there is a basis $\mathcal{B}'$ and some $\hat{\theta} \in S_\theta$ such that $G^{-1}_{\bullet\mathcal{B}'}(\triangle Q\hat{\theta}) \geq -G^{-1}_{\bullet\mathcal{B}'}q$. Clearly in this case $(\theta, \phi) = (\hat{\theta}, 0)$ is feasible to $LP_S(\mathcal{B}')$ and thus the optimal value must be nonpositive.

($\Leftarrow$): Recall that $LP_S(\mathcal{B}^*)$ has a feasible solution in which $\phi = 1$. Thus, since mpLCP (4.1) is equivalent to mpLCP^{ph1} with ϕ fixed to 0, if there exists a basis $\mathcal{B}'$ such that $LP_S(\mathcal{B}')$ is feasible for some $\hat{\phi} \leq 0$ then, since $\mathcal{K}(M)$ is convex when M is Q_0, there must exist a basis $\mathcal{B}''$ such that $LP_S(\mathcal{B}'')$ is feasible at $\phi = 0$. Therefore mpLCP must be feasible. □

For a given complementary basis $\mathcal{B}$, let $(\theta^*_{\mathcal{B}}, \phi^*_{\mathcal{B}})$ denote the optimal solution of $LP_S(\mathcal{B})$ and define

$$EQ_{\mathcal{B}} := \left\{ i \in \mathcal{B} : G^{-1}_{\bullet\mathcal{B}}(\triangle Q\theta^*_{\mathcal{B}} + r\phi^*_{\mathcal{B}}) = -G^{-1}_{\bullet\mathcal{B}}q \right\} \tag{4.53}$$

which is the set of indices in $\mathcal{B}$ whose corresponding defining constraints of $\mathcal{IR}^{ph1}_{\mathcal{B}}$ are binding at $(\theta^*_{\mathcal{B}}, \phi^*_{\mathcal{B}})$.

Proposition 4.45. Assume that M is a Q_0 matrix. Let a complementary basis $\mathcal{B}$ be given and let $(\theta^*_{\mathcal{B}}, \phi^*_{\mathcal{B}})$ represent the optimal solution of $LP_S(\mathcal{B})$ (4.52). Suppose that there does not exist an $i \in EQ_{\mathcal{B}}$ such that a diagonal or exchange pivot can be made from $\mathcal{B}$ which involves index i. Then the following hold:

- If $\phi^*_{\mathcal{B}} > 0$, then mpLCP is infeasible.
- If $\phi^*_{\mathcal{B}} = 0$ and $dim\,(\mathcal{IR}_{\mathcal{B}}) < k$, then there does not exist a feasible complementary basis $\mathcal{B}'$ such that $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k$.

Proof. If no diagonal or exchange pivots are possible which involve a particular index $i \in \mathcal{B}$, this indicates that the facet $cone\left(G_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ of the complementary cone $\mathcal{C}(\mathcal{B})$ forms a boundary of $\mathcal{K}(M)$. Thus, since $\mathcal{K}(M)$ is convex when M is a Q_0 matrix, all phase 1 invariancy regions lie in the same half-space defined by the hyperplane $(\hbar^i_{\mathcal{B}})^{ph1}$ that $\mathcal{IR}^{ph1}_{\mathcal{B}}$ lies in. Since this is true for all indices in $EQ_{\mathcal{B}}$, we have the following:

1. If the optimal value of $LP_S(\mathcal{B})$ is strictly positive, no phase 1 invariancy region exists which intersects the hyperplane $\phi = 0$.
2. If the optimal value of $LP_S(\mathcal{B})$ is zero, no phase 1 invariancy region other than $\mathcal{IR}^{ph1}_{\mathcal{B}}$ can have a nonempty intersection with the hyperplane $\phi = 0$.

Since any invariancy region $\mathcal{IR}_{\mathcal{B}}$ for mpLCP (4.1) is precisely the intersection of the hyperplane $\phi = 0$ and $\mathcal{IR}^{ph1}_{\mathcal{B}}$, the claim of the proposition follows. □

Observe that Proposition 4.45 provides the following two simplifications of the brute force method: (i) it identifies a subset of the facets of an invariancy region which need to be checked for adjacent invariancy regions, and (ii) it provides a stopping criterion under which one may conclude that either the mpLCP (4.1) is infeasible or there do not exist any full dimensional invariancy regions for mpLCP.

We now develop stopping criteria under which we can conclude that an initial basis with an invariancy region whose dimension is at least $k-1$ has been obtained. One such criterion is readily available. Suppose that for some complementary basis $\mathcal{B}$, the optimal value of $LP_S(\mathcal{B})$ is nonpositive and the optimal value of $LP_D(\mathcal{B})$ is strictly positive, then clearly $dim\,(\mathcal{IR}_{\mathcal{B}}) = k$. Another criterion is provided in the following proposition.

Proposition 4.46. Let a complementary basis $\mathcal{B}$ be given such that $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}}\right) \geq k$ and the optimal value of $LP_S(\mathcal{B})$ is nonnegative. Suppose that a new basis $\mathcal{B}'$ is obtained by either a diagonal or exchange pivot from $\mathcal{B}$ and that: (i) $\mathcal{IR}^{ph1}_{\mathcal{B}}$ and $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ are adjacent, and (ii) the optimal value of $LP_S(\mathcal{B}')$ is strictly negative. Then $dim\,(\mathcal{IR}_{\mathcal{B}'}) \geq k-1$.

Proof. Since $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}}\right) \geq k$ we know from the results of Corollaries 4.39 and 4.40 (and the fact that mpLCP$_{ph1}$ is a special case of mpLCP in which $k = k+1$) that $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}'}\right) \geq k$. Also, since $\mathcal{IR}^{ph1}_{\mathcal{B}}$ and $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ are adjacent, there exists $(\hat{\theta}, \hat{\phi}) \in \mathcal{IR}^{ph1}_{\mathcal{B}'}$ such that $\hat{\phi} > 0$. Since the optimal value of $LP_S(\mathcal{B}')$ is strictly negative, there also exists $(\theta', \phi') \in \mathcal{IR}^{ph1}_{\mathcal{B}'}$ such that $\phi' < 0$. This shows that the hyperplane $\phi = 0$ intersects $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ in its relative interior. Thus, since the hyperplane $\phi = 0$ is k-dimensional and $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ is at least k-dimensional, their intersection, which is precisely $\mathcal{IR}_{\mathcal{B}'}$, is at least $(k-1)$–dimensional. □

We are now able to use the results presented in this section to develop a more sophisticated strategy for obtaining the initial basis $\mathcal{B}_0$ such that $dim\,(\mathcal{IR}_{\mathcal{B}_0}) \geq k-1$. This strategy is outlined in Algorithm 4.2, which is presented in Section 4.5.

Before proceeding to Section 4.5, we return to the examples presented in Section 4.3 and briefly discuss how the theory presented here can be used to obtain initial bases. Observe $LP_S(\mathcal{B}_0)$ for each example:

$$\begin{array}{ll} \min\limits_{\theta,\phi} & \phi \\ \text{s.t.} & \theta_1 \geq -6 \\ & -\theta_2 + 2\phi \geq 1 \\ & \theta_1 - 3\theta_2 \geq -1 \\ & \theta \in [-2,2]^2 \end{array} \qquad\qquad \begin{array}{ll} \min\limits_{\theta,\phi} & \phi \\ \text{s.t.} & \theta_1 + \theta_2 + 4\phi \geq 3 \\ & -\theta_1 + 3\theta_2 \geq -1 \\ & \phi \geq 0 \\ & \theta \in [-2,2]^2 \end{array}$$

The optimal solutions of these LPs are $(\phi^*, \theta_1^*, \theta_2^*) = (-\frac{1}{2}, 0, -2)$ and $(\phi^{**}, \theta_1^{**}, \theta_2^{**}) = (0, \frac{3}{2}, \frac{3}{2})$, respectively. Hence, Proposition 4.44 implies that both instances of mpLCP are feasible, and as verified in Section 4.3, $\mathcal{IR}_{\mathcal{B}_0}$ is full dimensional for both instances. In these cases, finding an initial feasible basis with invariancy regions of dimension at least $k-1=1$ is quite easy. However, consider Example 2 and make the following simple modification. Assume that $S_\theta = [-2,1]^2$ rather than $S_\theta = [-2,2]^2$. It can be easily verified that in this case the optimal solution to $LP_S(\mathcal{B}_0)$ is $(\phi^{**}, \theta_1^{**}, \theta_2^{**}) = (\frac{1}{4}, 1, 1)$. This shows that $\mathcal{IR}^{ph1}_{\mathcal{B}_0}$ does not intersect the hyperplane $\phi = 0$ and thus $\mathcal{B}_0$ is infeasible for mpLCP. From (4.53) we find that $EQ_{\mathcal{B}_0} = \{w_1\}$. Hence, w_1 is the candidate for pivoting out of basis $\mathcal{B}_0$ which is most likely to reveal a new basis with an invariancy region that intersects the hyperplane $\phi = 0$. We know from Section 4.3 that this pivot will reveal $\mathcal{B}_1^{ex2} = \{z_1, w_2, w_3\}$. Consider $LP_S\left(\mathcal{B}_1^{ex2}\right)$:

$$\begin{array}{ll} \min\limits_{\theta,\phi} & \phi \\ \text{s.t.} & -\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 - \frac{4}{5}\phi \geq -\frac{3}{5} \\ & -\frac{9}{5}\theta_1 + \frac{11}{5}\theta_2 - \frac{16}{5}\phi \geq -\frac{17}{5} \\ & \frac{1}{5}\theta_1 + \frac{1}{5}\theta_2 + \frac{9}{5}\phi \geq \frac{3}{5} \\ & \theta \in [-2,1]^2 \end{array}$$

This LP has solution $(\phi^{**}, \theta_1^{**}, \theta_2^{**}) = (\frac{1}{9}, 1, 1)$, which shows that $EQ_{\mathcal{B}_1^{ex2}} = \{w_3\}$. Again, from Section 4.3, we know that pivoting on w_3 results in $\mathcal{B}_2^{ex2} = \{z_1, w_2, z_3\}$. Consider $LP_S\left(\mathcal{B}_2^{ex2}\right)$:

$$\begin{array}{ll} \min\limits_{\theta,\phi} & \phi \\ \text{s.t.} & -\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 - \frac{4}{5}\phi \geq -\frac{3}{5} \\ & -\frac{9}{5}\theta_1 + \frac{11}{5}\theta_2 - \frac{16}{5}\phi \geq -\frac{17}{5} \\ & -\frac{1}{5}\theta_1 - \frac{1}{5}\theta_2 - \frac{9}{5}\phi \geq -\frac{3}{5} \\ & \theta \in [-2,1]^2 \end{array}$$

This LP unbounded and thus, by Proposition 4.46, $\mathcal{B}_2^{ex2}$ serves as the initial basis for Example 2.

We have now shown how the theory from this section can be applied to an instance of mpLCP. A detailed algorithm showing explicilty how these steps should be applied, and in what order, is provided in Section 4.5.

4.5 The Algorithms and their Performance

This section consists of two subsections. In the first, we present Algorithms 4.1 and 4.2 which are designed for partitioning the parameter space S_θ and obtaining an initial feasible basis $\mathcal{B}_0$ such that $dim\left(\mathcal{IR}_{\mathcal{B}_0}\right) \geq k-1$, respectively. In the second subsection we discuss the complexity and performance of Algorithms 4.1 and 4.2.

4.5.1 Presentation of the Algorithms

In these algorithms we use the notation:

- $LCP(\hat{\theta})$ – the nonparametric LCP resulting from fixing θ to $\hat{\theta} \in S_\theta$ in (4.1).
- $\mathscr{R}$ – the set of invariancy regions for which adjacent regions need to be found.
- $\mathscr{K}$ – the set of feasible bases discovered.
- $\mathscr{F}_{\mathscr{B}}$ – for a given basis $\mathscr{B}$, the set of $i \in \mathscr{B}$ for which $\mathscr{h}^i_{\mathscr{B}}$ has been shown to form a facet of $\mathcal{IR}_{\mathscr{B}}$.
- $\mathscr{P}_{\mathscr{B}}$ – the subset of $\mathscr{F}_{\mathscr{B}}$ which contains all $i \in \mathscr{B}$ for which the facet of $\mathcal{IR}_{\mathscr{B}}$ formed by $\mathscr{h}^i_{\mathscr{B}}$ is known to also form a facet of $\mathcal{IR}_{\mathscr{B}'}$ for one and only one $\mathscr{B}' \in \mathscr{K}$.
- $\mathscr{L}_{\mathscr{B}}$ – for a given basis $\mathscr{B}$, the set of indices in $\mathscr{B}$ which cause $\mathcal{IR}_{\mathscr{B}}$ to be $k-1$ dimensional. In general, $\mathscr{L}_{\mathscr{B}} = \mathscr{L}^i_{\mathscr{B}',\mathscr{B}} \cup \left(\bigcup_{j \in \mathscr{L}^i_{\mathscr{B}',\mathscr{B}}} \mathrm{H}^j_{\mathscr{B}} \right)$, where $\mathscr{B}'$ is the basis from which a pivot was made in order to discover basis $\mathscr{B}$ for the first time and $i \in \mathscr{B}'$ is the index such that $\mathcal{IR}_{\mathscr{B}'}$ and $\mathcal{IR}_{\mathscr{B}}$ are adjacent along $\mathscr{h}^i_{\mathscr{B}'}$. Notice that if $\mathscr{L}^i_{\mathscr{B}',\mathscr{B}} = \emptyset$ then $\mathscr{L}_{\mathscr{B}} = \emptyset$ as well.
- $\mathrm{D}_{\mathscr{B}} := \left\{ i \in \mathscr{B} : (T_{\mathscr{B}})_{i,\bar{i}} \neq 0 \right\}$ (i.e., for basis $\mathscr{B}$, the set of indices in $\mathscr{B}$ which are candidates for a diagonal pivot.)
- $\mathrm{E}_{\mathscr{B}} := \left\{ (i,j) \in \mathscr{B} : i \neq j, (T_{\mathscr{B}})_{j,\bar{i}} > 0, (T_{\mathscr{B}})_{i,\bar{j}} \neq 0 \right\}$ (i.e., for basis $\mathscr{B}$, the set of pairs of indices in $\mathscr{B}$ which are candidates for an exchange pivot.)

Note that for that phase 1 problem, when appropriate, we will attach a superscript of $ph1$ to the above notations. We now present the algorithms, each followed by a brief description.

In Algorithm 4.1 we consider discovered invariancy regions one at a time. Full dimensional regions are processed in lines 3-15, while $(k-1)$–dimensional regions are processed in lines 16-23. For a full dimensional region we first need to determine its facets. This is done in lines 4-8. We construct sets of already known facets in lines 4-6 and determine any unknown facets in lines 7-8. We next determine bases which yield adjacent invariancy regions across each facet of interest. For full dimensional regions, this is done in lines 9-15, and for $(k-1)$–dimensional regions it is done in lines 17-23. Note that diagonal pivots are considered in lines 10-12 and 18-20, and exchange pivots are considered in lines 13-15 and 21-23.

A strategy for obtaining an initial basis $\mathscr{B}_0$ such that $dim\left(\mathcal{IR}_{\mathscr{B}_0}\right) \geq k-1$ is given in Algorithm 4.2. Note that Algorithm 4.2 is organized in a similar fashion to Algorithm 4.1 with

Algorithm 4.1 Partition S_θ.
Input: Sets $\mathcal{R}$ and $\mathcal{K}$ as well as an initial basis $\mathcal{B}_0$ and the corresponding sets $\mathcal{F}_{\mathcal{B}_0}$, $\mathcal{P}_{\mathcal{B}_0}$ and $\mathcal{L}_{\mathcal{B}_0}$.
Output: Sets $\mathcal{R}$ and $\mathcal{K}$. Upon termination, the invariancy regions contained in $\mathcal{R}$ partition S_θ.

```
 1: while $\mathcal{R} \neq \emptyset$ do
 2:     Remove an invariancy region $\mathcal{IR}_{\mathcal{B}}$ from $\mathcal{R}$.
 3:     if $\mathcal{L}_{\mathcal{B}} = \emptyset$ then                              ▷ i.e., $dim\,(\mathcal{IR}_{\mathcal{B}}) = k$
 4:         for $i \in \mathcal{F}_{\mathcal{B}}$ do
 5:             Set $\mathcal{F}_{\mathcal{B}} = \mathcal{F}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}}$.
 6:             if $i \in \mathcal{P}_{\mathcal{B}}$ then set $\mathcal{P}_{\mathcal{B}} = \mathcal{P}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}}$.
 7:         for $i \in (\mathcal{B} \setminus \mathcal{F}_{\mathcal{B}})$ do
 8:             if $LP_F(\mathcal{B}, i)$ has strictly positive optimal value then set $\mathcal{F}_{\mathcal{B}} = \mathcal{F}_{\mathcal{B}} \cup \left(\{i\} \cup \mathrm{H}^i_{\mathcal{B}}\right)$.
 9:         for $i \in \mathcal{F}_{\mathcal{B}} \setminus \mathcal{P}_{\mathcal{B}}$ do
10:             if $(T_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$ then
11:                 if $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\} \notin \mathcal{K}$ then add $\mathcal{IR}_{\mathcal{B}'}$ to $\mathcal{R}$, $\mathcal{B}'$ to $\mathcal{K}$, and $\bar{\imath}$ to $\mathcal{F}_{\mathcal{B}'}$ and $\mathcal{P}_{\mathcal{B}'}$. Set
                    $\mathcal{L}_{\mathcal{B}'} = \mathcal{L}^i_{\mathcal{B},\mathcal{B}'} \cup \left(\bigcup_{j \in \mathcal{L}^i_{\mathcal{B},\mathcal{B}'}} \mathrm{H}^j_{\mathcal{B}'}\right)$.
12:                 else if $\mathcal{IR}_{\mathcal{B}'} \in \mathcal{R}$ then add $\bar{\imath}$ to $\mathcal{P}_{\mathcal{B}'}$.
13:             else
14:                 for $j \in \mathcal{B}$ such that $(i, j) \in \mathrm{E}_{\mathcal{B}}$ and $\mathcal{B}'' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\} \notin \mathcal{K}$ do
15:                     if $LP_A(\mathcal{B}, i, j)$ has strictly positive optimal value then add $\mathcal{IR}_{\mathcal{B}''}$ to $\mathcal{R}$, $\mathcal{B}''$ to $\mathcal{K}$, and
                        $\bar{\imath}$ to $\mathcal{F}_{\mathcal{B}''}$. Set $\mathcal{L}_{\mathcal{B}''} = \mathcal{L}^i_{\mathcal{B},\mathcal{B}''} \cup \left(\bigcup_{\xi \in \mathcal{L}^i_{\mathcal{B},\mathcal{B}''}} \mathrm{H}^\xi_{\mathcal{B}''}\right)$.
16:     else
17:         for $\ell \in \mathcal{L}_{\mathcal{B}}$ do
18:             if $(T_{\mathcal{B}})_{\ell,\bar{\ell}} \neq 0$ then
19:                 if $\mathcal{B}' = (\mathcal{B} \setminus \{\ell\}) \cup \{\bar{\ell}\} \notin \mathcal{K}$ then add $\mathcal{IR}_{\mathcal{B}'}$ to $\mathcal{R}$, $\mathcal{B}'$ to $\mathcal{K}$, and $\bar{\ell}$ to $\mathcal{F}_{\mathcal{B}'}$ and $\mathcal{P}_{\mathcal{B}'}$. Set
                    $\mathcal{L}_{\mathcal{B}'} = \mathcal{L}^\ell_{\mathcal{B},\mathcal{B}'} \cup \left(\bigcup_{j \in \mathcal{L}^\ell_{\mathcal{B},\mathcal{B}'}} \mathrm{H}^j_{\mathcal{B}'}\right)$.
20:                 else if $\mathcal{IR}_{\mathcal{B}'} \in \mathcal{R}$ then add $\bar{\ell}$ to $\mathcal{P}_{\mathcal{B}'}$.
21:             else
22:                 for $j \in \mathcal{B}$ such that $(\ell, j) \in \mathrm{E}_{\mathcal{B}}$ and $\mathcal{B}'' = (\mathcal{B} \setminus \{\ell, j\}) \cup \{\bar{\ell}, \bar{\jmath}\} \notin \mathcal{K}$ do
23:                     if $LP_{A2}(\mathcal{B}, \ell, j)$ has strictly negative optimal value then add $\mathcal{IR}_{\mathcal{B}''}$ to $\mathcal{R}$, $\mathcal{B}''$ to $\mathcal{K}$,
                        and $\bar{\ell}$ to $\mathcal{F}_{\mathcal{B}''}$. Set $\mathcal{L}_{\mathcal{B}''} = \mathcal{L}^\ell_{\mathcal{B},\mathcal{B}''} \cup \left(\bigcup_{\xi \in \mathcal{L}^\ell_{\mathcal{B},\mathcal{B}''}} \mathrm{H}^\xi_{\mathcal{B}''}\right)$.
```

a few major exceptions. Lines 2 and 5-7 provide stopping criteria under which we can conclude that either mpLCP has no full dimensional invariancy region (and thus partitioning S_θ is futile), or that an initial basis has been found, respectively. In line 4 LP_S (4.52) is solved which serves two purposes: (i) the solution to LP_S is used to check the stopping criteria of lines 5-7, and (ii) if the stopping criteria are not satisfied, the solution to LP_S is used to determine the facets of a phase 1 invariancy region across which adjacent phase 1 invariancy regions may exist that have the potential to intersect the hyperplane $\phi = 0$. Full dimensional phase 1 regions are then processed in lines 10-16, while k–dimensional phase 1 regions are processed in lines 18-23. For full dimensional phase 1 regions we determine its facets which contain the point (θ^*, ϕ^*), the solution to LP_S. This is done in lines 10-11. We next determine bases which yield adjacent invariancy regions across each facet of interest. For full dimensional phase 1 regions, this is done in lines 12-16, and for k–dimensional

Algorithm 4.2 Find initial feasible complementary basis $\mathcal{B}_0$ with $dim\,(\mathcal{IR}_{\mathcal{B}_0}) = k$.
Input: An augmented parametric LCP problem $LCP(\theta, \phi)$ as in (4.50), defined by matrices M and $\triangle Q$ and vector q. A stack $\mathcal{S}$ of bases, initialized as $\mathcal{S} = \left\{\mathcal{B}^* = \{1, \ldots, h\}\right\}$. An empty set $\mathcal{B}_0 = \emptyset$.
Output: Initialized sets $\mathcal{R}$ and $\mathcal{K}$ and an initial basis $\mathcal{B}_0$ along with the set $\mathcal{L}_{\mathcal{B}_0}$.

```
 1: while $\mathcal{B}_0 = \emptyset$ do
 2:     if $\mathcal{S} = \emptyset$ then STOP. No full dimensional invariancy region exists for mpLCP (4.1).
 3:     else Pop $\mathcal{B}$ from $\mathcal{S}$.
 4:     Solve $LP_S(\mathcal{B})$ and obtain solution $(\theta^*_{\mathcal{B}}, \phi^*_{\mathcal{B}})$.
 5:     if $\phi^*_{\mathcal{B}} < 0$ then set $\mathcal{B}_0 = \mathcal{B}$, $\mathcal{R} = \{\mathcal{IR}_{\mathcal{B}_0}\}$, and $\mathcal{L}_{\mathcal{B}_0} = \mathcal{L}_{\mathcal{B}}$. STOP, an initial basis has been found.
 6:     else if $\phi^*_{\mathcal{B}} = 0$ then
 7:         if $LP_D(\mathcal{B})$ has a strictly positive optimal value then set $\mathcal{B}_0 = \mathcal{B}$, $\mathcal{R} = \{\mathcal{IR}_{\mathcal{B}_0}\}$, and $\mathcal{L}_{\mathcal{B}_0} = \mathcal{L}_{\mathcal{B}}$. STOP, an initial basis has been found.
 8:     else
 9:         if $\mathcal{L}_{\mathcal{B}} = \emptyset$ then
10:             for $i \in EQ_{\mathcal{B}}$ do
11:                 if $LP^{ph1}_F(\mathcal{B})$ has a strictly positive optimal value then
12:                     if $(T^{ph1}_{\mathcal{B}})_{i,\bar{i}} \neq 0$ then
13:                         if $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{i}\} \notin \mathcal{K}$ then add $\mathcal{B}'$ to $\mathcal{S}$ and $\mathcal{K}$. Set $\mathcal{L}_{\mathcal{B}'} = \mathcal{L}^i_{\mathcal{B},\mathcal{B}'} \cup \left(\bigcup_{j \in \mathcal{L}^i_{\mathcal{B},\mathcal{B}'}} \mathrm{H}^j_{\mathcal{B}'}\right)$.
14:                     else
15:                         for $j \in \mathcal{B}$ such that $(i,j) \in \mathrm{E}_{\mathcal{B}}$ and $\mathcal{B}'' = (\mathcal{B} \setminus \{i,j\}) \cup \{\bar{i}, \bar{j}\} \notin \mathcal{K}$ do
16:                             if $LP^{ph1}_A(\mathcal{B}, i, j)$ has strictly positive optimal value then add $\mathcal{B}''$ to $\mathcal{S}$ and $\mathcal{K}$. Set $\mathcal{L}_{\mathcal{B}''} = \mathcal{L}^i_{\mathcal{B},\mathcal{B}''} \cup \left(\bigcup_{\xi \in \mathcal{L}^i_{\mathcal{B},\mathcal{B}''}} \mathrm{H}^{\xi}_{\mathcal{B}''}\right)$.
17:         else
18:             for $i \in \left(\mathcal{L}_{\mathcal{B}} \cap EQ_{\mathcal{B}}\right)$ do
19:                 if $(T^{ph1}_{\mathcal{B}})_{i,\bar{i}} \neq 0$ then
20:                     if $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{i}\} \notin \mathcal{K}$ then add $\mathcal{B}'$ to $\mathcal{S}$ and $\mathcal{K}$. Set $\mathcal{L}_{\mathcal{B}'} = \mathcal{L}^{\ell}_{\mathcal{B},\mathcal{B}'} \cup \left(\bigcup_{j \in \mathcal{L}^{\ell}_{\mathcal{B},\mathcal{B}'}} \mathrm{H}^j_{\mathcal{B}'}\right)$.
21:                 else
22:                     for $j \in \mathcal{B}$ such that $(i,j) \in \mathrm{E}_{\mathcal{B}}$ and $\mathcal{B}'' = (\mathcal{B} \setminus \{i,j\}) \cup \{\bar{i}, \bar{j}\} \notin \mathcal{K}$ do
23:                         if $LP^{ph1}_{A2}(\mathcal{B}, i, j)$ has strictly positive optimal value then add $\mathcal{B}''$ to $\mathcal{S}$ and $\mathcal{K}$. Set $\mathcal{L}_{\mathcal{B}''} = \mathcal{L}^i_{\mathcal{B},\mathcal{B}''} \cup \left(\bigcup_{\xi \in \mathcal{L}^i_{\mathcal{B},\mathcal{B}''}} \mathrm{H}^{\xi}_{\mathcal{B}''}\right)$.
```

phase 1 regions it is done in lines 19-23. Note that diagonal pivots are considered in lines 12-13 and 19-20, and exchange pivots are considered in lines 14-16 and 21-23.

4.5.2 Complexity and Performance

We now examine the complexity and performance of Algorithms 4.1 and 4.2. We first point out that the correctness of these algorithms is implied by the proofs of the propositions and corollaries presented in Sections 4.3 and 4.4. Also, both algorithms are finite since there is always a finite number of complementary bases for any LCP.

4.5.2.1 Complexity

We first consider the complexity of Algorithm 4.1. Since the number of possible complementary bases is exponential in h, there cannot exist a polynomial algorithm for partioning S_θ. Thus, given an invariancy region, we consider the complexity of determining its facets and finding all adjacent invariancy regions. For the sake of comparison, we use the following notation of Columbano et al. [18]: $T_{LP}(var, eq)$ represents the time to solve an LP in standard form with var and eq denoting the number of variables and equations, respectively. Consider the following proposition.

Proposition 4.47. If M is sufficient then for any complementary basis $\mathcal{B}$, the time needed to determine the facets of $\mathcal{IR}_\mathcal{B}$ and find all invariancy regions adjacent to $\mathcal{IR}_\mathcal{B}$ is at most:

$$hT_{LP}(h, k+1) + \frac{2h^2 - 3h + 1}{2} T_{LP}(2h-2, k+1) \tag{4.54}$$

Furthermore, in the case in which the affine space $\mathcal{AS}$ lies in general position with respect to each complementary cone, the above expression reduces to:

$$hT_{LP}(h, k+1) + \frac{h^2 - h}{2} T_{LP}(2h-2, k+1) \tag{4.55}$$

Proof. Recognize that the majority of the computation in Algorithm 4.1 occurs on lines 8, 15, and 23. In line 8, LP_F (4.27) is considered. If the dual of this LP is solved, the problem has h variables and $k+1$ constraints. From line 7 it is clear that for each basis this LP is solved at most h times. Next consider the LPs LP_A (4.28) and LP_{A2} (4.49) on lines 15 and 23, respectively. Again, if the dual problems of each are solved, the problems will have $2h-2$ variables and $k+1$ constraints. Since adjacent bases can differ by at most two elements, LP_A will be called at most $\frac{h^2-h}{2}$ times. Notice that LP_{A2} is only called if $\mathcal{L}_\mathcal{B} \neq \emptyset$ (i.e, $\mathcal{IR}_\mathcal{B}$ is $k-1$ dimensional). From (4.23) and lines 11, 15, 19, and 23 recognize that $\mathcal{L}_\mathcal{B}$ can contain at most h elements. Notice, however, that if $\mathcal{L}_\mathcal{B}$ contains exactly h elements, then $\mathcal{IR}_\mathcal{B}$ contains an entire $(k-1)$–dimensional hyperplane. This can only happen in the trivial case in which $\mathcal{AS}$ is completely contained in one complementary cone, i.e., there is only one invariancy region and it is full dimensional. Thus, we assume that this is not the case and that $\mathcal{L}_\mathcal{B}$ contains at most $h-1$ elements. Next notice that LP_{A2} may be called at most $h-1$ times for each element of $\mathcal{L}_\mathcal{B}$. Thus, LP_{A2} is called at most $\frac{(h-1)^2}{2}$ times for each basis $\mathcal{B}$. In all, this shows that the time needed to determine the facets of $\mathcal{IR}_\mathcal{B}$ and find all invariancy regions adjacent to $\mathcal{IR}_\mathcal{B}$ is at most (4.54).

If the affine space $\mathcal{AS}$ lies in general position with respect to each complementary cone then no invariancy region will be $k-1$ dimensional and thus LP_{A2} is never called. Therefore, in this case, (4.54) reduces to (4.55). $\square$

The result shown in Proposition 4.47 matches the result from [18] for the case in which $\mathcal{AS}$ lies in general position, but improves upon the reported complexity of $(h^2+h)T_{LP}(h,k+1) + \dfrac{h^3-h}{2}T_{LP}(2h,k+1)$ for the case in which $\mathcal{AS}$ does not lie in general position.

We now consider the performance of Algorithm 4.2. In [18] an initial basis and corresponding full-dimensional invariancy region are assumed to be given as input to their algorithm, but discussion is not provided as to how such an initial basis should be discovered. In this work, Algorithm 4.2 is used for this purpose. In the worst case this algorithm may explore every complementary basis. Thus, since the number of possible complementary bases is exponential in h, the worst-case complexity of the algorithm cannot be polynomial. Hence, given a phase 1 invariancy region, we consider the complexity of determining a desired subset of its facets and finding all adjacent invariancy regions across these facets.

Proposition 4.48. If M is sufficient then for any complementary basis $\mathscr{B}$, the time needed to determine a desired subset of the facets of $\mathcal{IR}_{\mathscr{B}}^{ph1}$ and find all invariancy regions adjacent to $\mathcal{IR}_{\mathscr{B}}^{ph1}$ across these facets is at most:

$$T_{LP}(h,k+1) + hT_{LP}(h,k+2) + \frac{2h^2-3h+1}{2}T_{LP}(2h-2,k+2)$$

Furthermore, in the case in which the affine space $\mathscr{AS}^{ph1} := \{q + \triangle Q\theta + r\phi : (\theta,\phi) \in S_\theta \times \mathbb{R}\}$ lies in general position with respect to each complementary cone, the above expression reduces to:

$$T_{LP}(h,k+1) + hT_{LP}(h,k+2) + \frac{h^2-h}{2}T_{LP}(2h-2,k+2)$$

Proof. Algorithm 4.2 is a special case of Algorithm 4.1 with two major exceptions: (i) there is one additional parameter, and (ii) an extra LP is solved for each basis. The given complexities are computed as extensions of those presented in Proposition 4.47 by: (i) replacing $k+1$ with $k+2$, and (ii) adding the term $T_{LP}(h,k+1)$ to account for solving LP_S on line 4. $\square$

4.5.2.2 Performance - Experimental Results

We now present the results of a computational experiment we conducted in order to test the practical performance of the proposed algorithms. We also include a few brief notes on our implementation.

We implemented the proposed two-phase algorithm using the C programming language and used the CPLEX optimization package to solve all auxiliary LPs. As we have already proved that the complexity of the proposed technique is lower than that of Columbano et al. [18], we chose to compare with the multiparametric toolbox (MPT) software package [41], available for MATLAB. Note that the MPT utilizes an implementation of the mpLCP solver proposed in [46], and although the package is run using MATLAB, the underlying implementation of the mpLCP method is written in the C programming language. We also used CPLEX as the default solver for LPs and QPs within the MPT (all other settings and parameters were left at their default values). All tests were run using MATLAB R2013b [61], MPT 3.0.20, and CPLEX 12.6 [45] on a machine running Linux Mint 16 with two 2.4GHz processors, each with 4GB of RAM.

For our experiment we utilized 49 QP instances made available by [59] and 11 QP instances from [79]. Each instance used is a convex QP, although some of the instances from [79] contained integer variables. In these cases we relaxed the integrality constraints. All instances were reformulated as LCPs and were then each converted to two distinct mpLCPs by randomly generating two matrices $\triangle Q$, one for $k = 2$ and one for $k = 3$. All randomly generated elements were taken from the closed interval [0,10]. Additionally, we set $S_\theta = [-2, 2]^k$ for each instance. (The data associated with these instances is available at `http://mthsc.clemson.edu/files/adelgren-instances.zip`.) We then solved each instance using our proposed method as well as the MPT. A summary of the results is given in Table 4.1. A complete version of the results is available in Tables 4.2 and 4.3, which can be found in Section 4.B. Recognize that in Table 4.1 we display the average time per discovered region (in seconds) rather than average overall time. There are two primary reasons for this: (i) we only allowed the MPT and the proposed method to work on an instance for a maximum of one hour (and report the number of regions computed in this time), and (ii) even for completely solved instances, the number of invariancy regions discovered by the MPT and the proposed method occasionally varied significantly. There are two main reasons for this variation: (i) numerical inconsistency arising from the fact that invariancy regions can be arbitrarily small (for visual evidence of this, observe

Table 4.1: Experimental Results - MPT results are displayed in standard font; results from the proposed method are in bold.

k	h	Num. Instances	Num. Failed		$<$ 1 hr.		$\geq$ 1 hr.		Avg. Time/Rgn (s)		Avg. Num. Rgns		Avg. Ph1 Iter	Avg. Ph2 Iter
2	$(0, 50)$	14	0	**0**	14	**14**	0	**0**	0.195	**0.002**	24	**29**	**2**	**30**
	$[50, 250)$	12	3	**3**	9	**9**	0	**0**	1.09	**0.02**	2,047	**8,687**	**1**	**9,035**
	$[250, 500)$	12	2	**0**	10	**10**	0	**2**	1.59	**0.07**	944	**7,349**	**156**	**9,385**
	$[500, 1000)$	17	7	**1**	5	**8**	5	**4**	5.34	**0.26**	3,104	**10,575**	**52**	**11,282**
	$[1000, 1500)$	5	0	**0**	1	**2**	4	**3**	30.02	**0.62**	373	**3,848**	**41**	**4,945**
3	$(0, 50)$	14	0	**0**	14	**14**	0	**0**	0.304	**0.002**	113	**116**	**2**	**116**
	$[50, 250)$	12	5	**3**	4	**5**	3	**4**	1.37	**0.04**	3,980	**37,467**	**2**	**38,401**
	$[250, 500)$	12	6	**0**	4	**6**	2	**6**	1.71	**0.17**	528	**16,491**	**122**	**18,242**
	$[500, 1000)$	17	5	**1**	1	**3**	11	**13**	9.20	**0.63**	1,798	**8,248**	**31**	**8,893**
	$[1000, 1500)$	5	0	**0**	0	**0**	5	**5**	34.14	**1.13**	54	**3,300**	**14**	**4,138**

Figure 4.8 in Appendix B), and (ii) on a small number of instances the MPT terminated early revealing a nonconvex, and thus incorrect, union of invariancy regions. In addition to this, there were six instances for $k = 2$ and two instances for $k = 3$ which the MPT claimed to be infeasible even though all instances were feasible by construction.

There are a few key pieces of information to observe from Table 4.1. First, the proposed algorithm fails on fewer of these instances than does the MPT. Second, the proposed algorithm is able to solve more instances in under an hour than the MPT. Finally, the average time per region is significantly lower for the proposed algorithm than the MPT.

We now discuss a few details of our implementation. First, we point out that our implementation is purely serial in nature, we even set the number of threads available to CPLEX to be one. We do not know, however, the number of threads used by the MPT or CPLEX when used from within MATLAB. We also point out that in our implementation we set the CPLEX feasibility tolerance to 10^{-7}, we used a tolerance of 10^{-5} when testing equality of values, and when solving auxiliary LPs we assumed that the variable λ was acceptably large when it exceeded a value of 10^{-6}. Additionally, in our implementation we explicitly compute the tableau associated with each discovered basis and thus the overall performance could likely be improved by instead using matrix factorization techniques. We also note that during testing we discovered that although phase 1 of the proposed algorithm is correct and finite, it can occasionally require a significant number of iterations. We found that this frequently occurred when a basis $\mathscr{B}$ was discovered for which the optimal solution to $LP_s(\mathscr{B})$ was zero, but $dim(\mathcal{IR}_{\mathscr{B}}) \neq k$. Oftentimes, if this occurred, a significant number of subsequent bases would be discovered which had the same property. We believe that this phenomena takes place primarily in the case in which the affine subspace $\mathscr{AS}$ contains the point $q = \mathbf{0}$ and hence intersects the common origin point of every complementary cone. In this situation, phase 1 can take significant time to discover a cone having a k-dimensional intersection with $\mathscr{AS}$

because adjacent bases are not explored in any particular order. To counter this, we propose the following improvement to phase 1, which we found to work quite well empirically. For each facet of a given phase 1 invariancy region which contains the optimal solution to LP_s, compute the normalized normal vector and then consider these facets on lines 10 and 18 of Algorithm 4.2 in ascending order of the ϕ component of these normalized vectors. This ensures that the last bases placed on stack $\mathcal{S}$ (and thus also the first bases taken off the stack) are those that have phase 1 invariancy regions containing a facet with a relatively large ϕ component of their normalized normal vectors. In this way phase 1 can more rapidly discover a phase 1 invariancy region containing a facet whose normalized ϕ component is equal to one, or has a k-dimensional intersection with the hyperplane $\phi = 0$.

4.6 Conclusion

In this work we have introduced a new two-phase method for solving mpLCP (4.1) in which M is a sufficient matrix. Phase 1 answers the previously unanswered question of how one can determine an initial full dimensional invariancy region which can be used as a starting point in the process of partitioning the parameter space S_θ. The partition of S_θ is carried out in Phase 2, which is inspired by the method introduced by Columbano et al. [18]. The worst-case complexity of the proposed two-phase method is $O\left(h^2 T_{LP}(h,k)\right)$, which improves upon the worst-case complexity of $O\left(h^3 T_{LP}(h,k)\right)$ reported in [18]. Here $T_{LP}(var, eq)$ represents the time to solve an LP in standard form with var and eq denoting the number of variables and equality constraints, respectively. Experimental results are provided which give strong evidence of the utility of the proposed methods. In Appendix A we provide a detailed example, showing the utilization of the proposed method.

In the future we aim to extend this method and develop a technique for solving mpLCP with parameters in general locations. Developing such a tool would have a variety of benefits, expanding well beyond the scope of LCP.

4.A Appendix A: An Example

Here we provide a detailed example of solving an instance of mpLCP and partitioning S_θ using Algorithms 4.1 and 4.2. Consider the following mpLCP:

$$w - \begin{bmatrix} 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & -2 & -4 \\ 0 & 0 & 0 & 0 & -1 \\ -2 & 2 & 0 & 4 & 1 \\ 1 & 4 & 1 & -1 & 2 \end{bmatrix} z = \begin{bmatrix} 4 \\ 16 \\ 4 \\ 4 \\ -8 \end{bmatrix} + \begin{bmatrix} 0 & -1 \\ 0 & 1 \\ 0 & -\frac{1}{2} \\ -2 & 0 \\ \frac{1}{2} & 0 \end{bmatrix} \theta \tag{4.56}$$
$$w^\top z = 0$$
$$w, z \geq 0$$

Assume that $S_\theta = [-4, 3] \times [-2, 2]$.

We now show how to partition S_θ. For the sake of clarity, throughout this example we use variable names to describe the elements of each basis rather than the corresponding indices. We begin with Phase 1.

<u>Phase 1: Initialization</u>:

To begin, we create mpLCP$_{ph1}$:

$$w - \begin{bmatrix} 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & -2 & -4 \\ 0 & 0 & 0 & 0 & -1 \\ -2 & 2 & 0 & 4 & 1 \\ 1 & 4 & 1 & -1 & 2 \end{bmatrix} z = \begin{bmatrix} 4 \\ 16 \\ 4 \\ 4 \\ -8 \end{bmatrix} + \begin{bmatrix} 0 & -1 \\ 0 & 1 \\ 0 & -\frac{1}{2} \\ -2 & 0 \\ \frac{1}{2} & 0 \end{bmatrix} \theta + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 9 \end{bmatrix} \phi \tag{4.57}$$
$$w^\top z = 0$$
$$w, z \geq 0$$

By construction, basis $\mathcal{B}^* = \{1, 2, 3, 4, 5\}$ is feasible and so we initialize the stack of bases $\mathcal{S}$ as $\mathcal{S} = \{\mathcal{B}^*\}$. Observe from (4.57) that:

$$\left(G^{-1}_{\bullet\mathcal{B}^*}\right)(q + \triangle Q\theta) = \begin{bmatrix} 4 - \theta_2 \\ 16 + \theta_2 \\ 4 - \frac{1}{2}\theta_2 \\ 4 - 2\theta_1 \\ -8 + \frac{1}{2}\theta_1 + 9\phi \end{bmatrix} \tag{4.58}$$

We proceed by removing $\mathcal{B}^*$ from $\mathcal{S}$ and solving $LP_S(\mathcal{B}^*)$:

$$\begin{array}{rc} \min\limits_{\theta,\phi} & \phi \\ \text{s.t.} & 4-\theta_2 \geq 0 \\ & 16+\theta_2 \geq 0 \\ & 4-\frac{1}{2}\theta_2 \geq 0 \\ & 4-2\theta_1 \geq 0 \\ & -8+\frac{1}{2}\theta_1+9\phi \geq 0 \\ & -4 \leq \theta_1 \leq 3 \\ & -2 \leq \theta_2 \leq 2 \end{array}$$

The optimal solution of this LP is $(\theta_1^*, \theta_2^*, \phi^*) = \left(2, 0, \frac{7}{9}\right)$, which shows that $EQ_{\mathcal{B}^*} = \{w_4, w_5\}$. Since the optimal value of $LP_S(\mathcal{B}^*)$ is strictly positive and $\mathcal{L}_{\mathcal{B}^*} = \emptyset$, we consider each element of $EQ_{\mathcal{B}^*}$ individually. To determine whether or not the hyperplanes $(\mathscr{h}_{\mathcal{B}^*}^{w_4})^{ph1}$ and $(\mathscr{h}_{\mathcal{B}^*}^{w_5})^{ph1}$ form facets of $\mathcal{IR}_{\mathcal{B}^*}^{ph1}$ we solve $LP_F^{ph1}(\mathcal{B}, w_4)$ and $LP_F^{ph1}(\mathcal{B}, w_5)$ (4.27). Observe these LPs:

$LP_F^{ph1}(\mathcal{B}, w_4)$:

$$\begin{array}{rc} \max\limits_{\lambda,\theta,\phi} & \lambda \\ \text{s.t.} & 4-\theta_2-\lambda \geq 0 \\ & 16+\theta_2-\lambda \geq 0 \\ & 4-\frac{1}{2}\theta_2-\lambda \geq 0 \\ & 4-2\theta_1 = 0 \\ & -8+\frac{1}{2}\theta_1+9\phi-\lambda \geq 0 \\ & -4 \leq \theta_1 \leq 3 \\ & -2 \leq \theta_2 \leq 2 \end{array}$$

$LP_F^{ph1}(\mathcal{B}, w_5)$:

$$\begin{array}{rc} \max\limits_{\lambda,\theta,\phi} & \lambda \\ \text{s.t.} & 4-\theta_2-\lambda \geq 0 \\ & 16+\theta_2-\lambda \geq 0 \\ & 4-\frac{1}{2}\theta_2-\lambda \geq 0 \\ & 4-2\theta_1-\lambda \geq 0 \\ & -8+\frac{1}{2}\theta_1+9\phi = 0 \\ & -4 \leq \theta_1 \leq 3 \\ & -2 \leq \theta_2 \leq 2 \end{array}$$

The optimal solution of $LP_F^{ph1}(\mathcal{B}, w_4)$ is $(\theta_1^*, \theta_2^*, \phi^*, \lambda^*) = \left(2, -2, \frac{4}{3}, 5\right)$ and the optimal solution of $LP_F^{ph1}(\mathcal{B}, w_5)$ is $(\theta_1^{**}, \theta_2^{**}, \phi^{**}, \lambda^{**}) = \left(-\frac{1}{2}, -2, \frac{11}{12}, 5\right)$. This shows that the optimal values of both LPs are strictly positive. Therefore we are interested in determining pivots from basis $\mathcal{B}^*$ which involve either w_4 or w_5. Without explicitly writing down $T_{\mathcal{B}^*}^{ph1}$ (4.18), one can observe from (4.57) that $(T_{\mathcal{B}^*}^{ph1})_{w_4,z_4} = -4$ and $(T_{\mathcal{B}^*}^{ph1})_{w_5,z_5} = -2$. Since these values are nonzero, we can perform diagonal pivots on these elements. Doing so provides two new bases, $\mathcal{B}' = \{w_1, w_2, w_3, z_4, w_5\}$ and $\mathcal{B}'' = \{w_1, w_2, w_3, w_4, z_5\}$. Observe their tableaux, in respective order:

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	0	0	$-\frac{1}{2}$	0	-1	1	0	0	$\frac{3}{2}$	$\theta_1 - \theta_2 + 2$
w_2	0	1	0	$\frac{1}{2}$	0	1	-1	0	0	$\frac{7}{2}$	$-\theta_1 + \theta_2 + 18$
w_3	0	0	1	0	0	0	0	0	0	1	$-\frac{1}{2}\theta_2 + 4$
z_4	0	0	0	$-\frac{1}{4}$	0	$-\frac{1}{2}$	$\frac{1}{2}$	0	1	$\frac{1}{4}$	$\frac{1}{2}\theta_1 - 1$
w_5	0	0	0	$-\frac{1}{4}$	1	$-\frac{3}{2}$	$-\frac{7}{2}$	-1	0	$-\frac{7}{4}$	$\theta_1 + 9\phi - 9$

(4.59)

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	0	0	0	$\frac{1}{2}$	$-\frac{1}{2}$	-2	$-\frac{1}{2}$	$-\frac{5}{2}$	0	$\frac{1}{4}\theta_1 - \theta_2 + \frac{9}{2}\phi$
w_2	0	1	0	0	2	-2	-8	-2	0	0	$\theta_1 + \theta_2 + 18\phi$
w_3	0	0	1	0	$\frac{1}{2}$	$-\frac{1}{2}$	-2	$-\frac{1}{2}$	$-\frac{1}{2}$	0	$\frac{1}{4}\theta_1 - \frac{1}{2}\theta_2 + \frac{9}{2}\phi$
w_4	0	0	0	1	$-\frac{1}{2}$	$\frac{5}{2}$	0	$\frac{1}{2}$	$-\frac{7}{2}$	0	$-\frac{9}{48}\theta_1 - \frac{9}{2}\phi + 8$
z_5	0	0	0	0	$-\frac{1}{2}$	$\frac{1}{2}$	2	$\frac{1}{2}$	$\frac{1}{2}$	1	$-\frac{1}{4}\theta_1 - \frac{9}{2}\phi + 4$

(4.60)

Since neither $\mathcal{B}'$ nor $\mathcal{B}''$ is in $\mathcal{K}$, we add each to both $\mathcal{S}$ and $\mathcal{K}$. Also, observe from (4.59) and (4.60) that $\mathrm{L}^{w_4}_{\mathcal{B}^*,\mathcal{B}'} = \emptyset$ and $\mathrm{L}^{w_5}_{\mathcal{B}^*,\mathcal{B}''} = \emptyset$. Thus, we set both $\mathcal{L}_{\mathcal{B}'}$ and $\mathcal{L}_{\mathcal{B}''}$ equal to the empty set.

We have now completed one iteration through the while loop of Algorithm 4.2 and are ready to begin a second iteration. We must select a basis from $\mathcal{S}$ to consider next. Since $\mathcal{S}$ is a stack, we use a last in, first out strategy for this selection. Thus, we now consider basis $\mathcal{B}''$ and solve $LP_S(\mathcal{B}'')$ (4.52):

$$\begin{array}{cc}
\min\limits_{\theta,\phi} & \phi \\
\text{s.t.} & \frac{1}{4}\theta_1 - \theta_2 + \frac{9}{2}\phi \geq 0 \\
& \theta_1 + \theta_2 + 18\phi \geq 0 \\
& \frac{1}{4}\theta_1 - \frac{1}{2}\theta_2 + \frac{9}{2}\phi \geq 0 \\
& -\frac{9}{48}\theta_1 - \frac{9}{2}\phi + 8 \geq 0 \\
& -\frac{1}{4}\theta_1 - \frac{9}{2}\phi + 4 \geq 0 \\
& -4 \leq \theta_1 \leq 3 \\
& -2 \leq \theta_2 \leq 2
\end{array}$$

The optimal solution of this LP is $(\theta_1^*, \theta_2^*, \phi^*) = \left(3, 0, -\frac{1}{6}\right)$. Since the optimal value is strictly negative we conclude that we have found an initial basis. Since $\mathcal{L}_{\mathcal{B}''} = \emptyset$ we also know that $\mathcal{IR}_{\mathcal{B}''}$ is full dimensional. We now set $\mathcal{B}_0 = \mathcal{B}''$, $\mathcal{R} = \{\mathcal{IR}_{\mathcal{B}_0}\}$ and $\mathcal{L}_{\mathcal{B}_0} = \mathcal{L}_{\mathcal{B}''}$ and exit Phase 1. For a visualization of the invariancy regions discovered during Phase 1, see Figure 4.6. Recall that $\mathcal{IR}_{\mathcal{B}_0}$ is precisely the intersection of $\mathcal{IR}^{\phi 1}_{\mathcal{B}_0}$ and the plane $\phi = 0$.

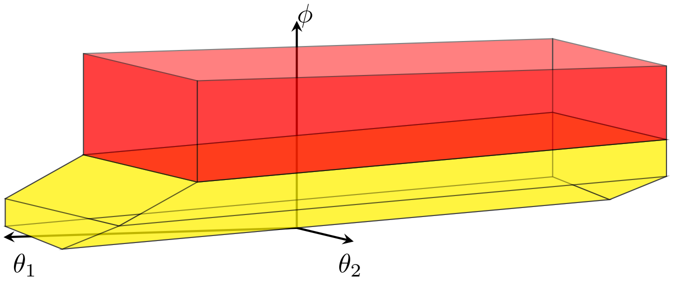


Figure 4.6: $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$ is shown in red and $\mathcal{IR}^{ph1}_{\mathcal{B}''}$ is shown in yellow.

<u>Phase 2</u>:

<u>Iteration 1</u>: Remove $\mathcal{IR}_{\mathcal{B}_0}$ from $\mathcal{R}$, resulting in $\mathcal{R} = \emptyset$. Also, we currently have $\mathcal{F}_{\mathcal{B}_0} = \mathcal{P}_{\mathcal{B}_0} = \emptyset$. Notice that $T_{\mathcal{B}_0}$ is the same as the tableau shown in (4.60) with the exception that $\phi = 0$. Now, since $\mathcal{L}_{\mathcal{B}_0} = \emptyset$ we consider each element of $\mathcal{B}_0$.

w_1 - Since $w_1 \notin \mathcal{F}_{\mathcal{B}_0}$ we will solve $LP_F(\mathcal{B}_0, w_1)$ (4.27). Notice from (4.60) that $\mathrm{Z}_{\mathcal{B}_0} = \mathrm{H}^{w_1}_{\mathcal{B}_0} = \emptyset$. Thus, $LP_F(\mathcal{B}_0, w_1)$ (4.27) has the form:

$$\begin{array}{cc} \max\limits_{\lambda,\theta} & \lambda \\ \text{s.t.} & \frac{1}{4}\theta_1 - \theta_2 = 0 \\ & \theta_1 + \theta_2 - \lambda \geq 0 \\ & \frac{1}{4}\theta_1 - \frac{1}{2}\theta_2 - \lambda \geq 0 \\ & -\frac{9}{48}\theta_1 - \lambda \geq -8 \\ & -\frac{1}{4}\theta_1 - \lambda \geq -4 \\ & -4 \leq \theta_1 \leq 3 \\ & -2 \leq \theta_2 \leq 2 \end{array}$$

The solution of this LP is $(\theta_1^*, \theta_2^*, \lambda^*) = (3, \frac{3}{4}, \frac{3}{8})$. Since the optimal value is positive, $\hbar^{w_1}_{\mathcal{B}_0}$ forms a facet of $\mathcal{IR}_{\mathcal{B}_0}$. Therefore, we set $\mathcal{F}_{\mathcal{B}_0} = \{w_1\}$.

w_2 - Since $w_2 \notin \mathcal{F}_{\mathcal{B}_0}$ we solve $LP_F(\mathcal{B}_0, w_2)$. The solution is $(\theta_1^*, \theta_2^*, \lambda^*) = (2, -2, \frac{3}{2})$, showing that the optimal value is positive. Thus, $\hbar^{w_2}_{\mathcal{B}_0}$ forms a facet of $\mathcal{IR}_{\mathcal{B}_0}$ and so we set $\mathcal{F}_{\mathcal{B}_0} = \mathcal{F}_{\mathcal{B}_0} \cup \{w_2\}$.

w_3 - Since $w_3 \notin \mathcal{F}_{\mathcal{B}_0}$ we solve $LP_F(\mathcal{B}_0, w_3)$. The solution is $(\theta_1^*, \theta_2^*, \lambda^*) = (0, 0, 0)$. Since the optimal value of $LP_F(\mathcal{B}_0, w_3)$ is not strictly positive, $\hbar^{w_3}_{\mathcal{B}_0}$ does not form a facet of $\mathcal{IR}_{\mathcal{B}_0}$.

z_4 - Since $w_4 \notin \mathcal{F}_{\mathcal{B}_0}$ we solve $LP_F(\mathcal{B}_0, w_4)$. This LP is infeasible, though, and thus $\hbar^{w_4}_{\mathcal{B}_0}$ does not form a facet of $\mathcal{IR}_{\mathcal{B}_0}$.

z_5 - Since $z_5 \notin \mathcal{F}_{\mathcal{B}_0}$ we solve $LP_F(\mathcal{B}_0, z_5)$. However, this LP is also infeasible and so $\mathcal{h}^{z_5}_{\mathcal{B}_0}$ does not form a facet of $\mathcal{IR}_{\mathcal{B}_0}$.

To see that the correct conclusions have been made about which hyperplanes form facets of $\mathcal{IR}_{\mathcal{B}_0}$, observe Figure 4.7a.

Since we have now determined the facets of $\mathcal{IR}_{\mathcal{B}_0}$, we next need to determine all invariancy regions which are adjacent to $\mathcal{IR}_{\mathcal{B}_0}$ along each facet. Thus, we consider each element of $\mathcal{F}_{\mathcal{B}_0} = \{w_1, w_2\}$.

w_1 - Notice from (4.60) that $(T_{\mathcal{B}_0})_{w_1,z_1} = -\frac{1}{2} \neq 0$. Since $(\mathcal{B}_0 \setminus \{w_1\}) \cup \{z_1\} \notin \mathcal{K}$, we let $\mathcal{B}_1 = (\mathcal{B}_0 \setminus \{w_1\}) \cup \{z_1\}$, $\mathcal{R} = \mathcal{R} \cup \mathcal{IR}_{\mathcal{B}_1}$, $\mathcal{K} = \mathcal{K} \cup \mathcal{B}_1$, $\mathcal{F}_{\mathcal{B}_1} = \mathcal{F}_{\mathcal{B}_1} \cup \{z_1\}$, and $\mathcal{P}_{\mathcal{B}_1} = \mathcal{P}_{\mathcal{B}_1} \cup \{z_1\}$. Notice that $T_{\mathcal{B}_1}$ can be obtained from $T_{\mathcal{B}_0}$ by performing a pivot on $(T_{\mathcal{B}_0})_{w_1,z_1}$:

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
z_1	-2	0	0	0	-1	1	4	1	5	0	$-\frac{1}{2}\theta_1 + 2\theta_2$
w_2	-4	1	0	0	0	0	0	0	10	0	$5\theta_2$
w_3	-1	0	1	0	0	0	0	0	2	0	$\frac{1}{2}\theta_2$
w_4	5	0	0	1	2	0	-10	-2	-16	0	$-\theta_1 - 5\theta_2 + 8$
z_5	1	0	0	0	0	0	0	0	-2	1	$-\theta_2 + 4$

(4.61)

Recognize from (4.61) and (4.48) that $\mathcal{L}^{w_1}_{\mathcal{B}_0,\mathcal{B}_1} = \emptyset$. Thus, we set $\mathcal{L}_{\mathcal{B}_1} = \emptyset$.

w_2 - Notice from (4.60) that $(T_{\mathcal{B}_0})_{w_2,z_2} = -8 \neq 0$. Since $(\mathcal{B}_0 \setminus \{w_2\}) \cup \{z_2\} \notin \mathcal{K}$, we let $\mathcal{B}_2 = (\mathcal{B}_0 \setminus \{w_2\}) \cup \{z_2\}$, $\mathcal{R} = \mathcal{R} \cup \mathcal{IR}_{\mathcal{B}_2}$, $\mathcal{K} = \mathcal{K} \cup \mathcal{B}_2$, $\mathcal{F}_{\mathcal{B}_2} = \mathcal{F}_{\mathcal{B}_2} \cup \{z_2\}$, and $\mathcal{P}_{\mathcal{B}_2} = \mathcal{P}_{\mathcal{B}_2} \cup \{z_2\}$. Notice that $T_{\mathcal{B}_2}$ can be obtained from $T_{\mathcal{B}_0}$ by performing a pivot on $(T_{\mathcal{B}_0})_{w_2,z_2}$:

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	$-\frac{1}{4}$	0	0	0	0	0	0	$-\frac{5}{2}$	0	$-\frac{5}{4}\theta_2$
z_2	0	$-\frac{1}{8}$	0	0	$-\frac{1}{4}$	$\frac{1}{4}$	1	$\frac{1}{4}$	0	0	$-\frac{1}{8}\theta_1 - \frac{1}{8}\theta_2$
w_3	0	$-\frac{1}{4}$	1	0	0	0	0	0	$-\frac{1}{2}$	0	$-\frac{3}{4}\theta_2$
w_4	0	0	0	1	$-\frac{1}{2}$	$\frac{5}{2}$	0	$\frac{1}{2}$	$-\frac{7}{2}$	0	$-\frac{9}{4}\theta_1 + 8$
z_5	0	$\frac{1}{4}$	0	0	0	0	0	0	$\frac{1}{2}$	1	$\frac{1}{4}\theta_2 + 4$

(4.62)

Recognize from (4.63) that $\mathcal{L}^{w_2}_{\mathcal{B}_0,\mathcal{B}_2} = \emptyset$. Thus, we set $\mathcal{L}_{\mathcal{B}_2} = \emptyset$.

This completes iteration 1.

<u>Iteration 2</u>: Remove $\mathcal{IR}_{\mathcal{B}_1}$ from $\mathcal{R}$, resulting in $\mathcal{R} = \{\mathcal{IR}_{\mathcal{B}_2}\}$. The tableau for $\mathcal{B}_1$ is shown in (4.61). Since $\mathcal{L}_{\mathcal{B}_1} = \emptyset$ we consider each element of $\mathcal{B}_1 \setminus \mathcal{F}_{\mathcal{B}_1}$ in order to determine the unknown facets of $\mathcal{IR}_{\mathcal{B}_1}$. Since the work here is analogous to work done during iteration 1, the details are omitted. We find that, in addition to $\mathscr{h}^{z_1}_{\mathcal{B}_1}$, the facets of $\mathcal{IR}_{\mathcal{B}_1}$ are formed by $\mathscr{h}^{w_2}_{\mathcal{B}_1}$, $\mathscr{h}^{w_3}_{\mathcal{B}_1}$ and $\mathscr{h}^{w_4}_{\mathcal{B}_1}$. Note that $\mathscr{h}^{w_2}_{\mathcal{B}_1} = \mathscr{h}^{w_3}_{\mathcal{B}_1}$.

w_2 - Notice from (4.61) that $(T_{\mathcal{B}_1})_{w_2,z_2} = 0$. Thus, a diagonal pivot cannot be performed on $(T_{\mathcal{B}_1})_{w_2,z_2}$. Instead, we look for possible exchange pivots. The only possible exchange pivot involves z_1 and w_2. However, since $(\mathcal{B}_1 \setminus \{z_1, w_2\}) \cup \{w_1, z_2\} = \mathcal{B}_2$, we do not make this pivot.

w_3 - Since $(T_{\mathcal{B}_1})_{w_3,z_3} = 0$, a diagonal pivot cannot be performed on $(T_{\mathcal{B}_1})_{w_3,z_3}$. The only possible exchange pivot involving w_3 also involves z_1. Since $(\mathcal{B}_1 \setminus \{z_1, w_3\}) \cup \{w_1, z_3\} \notin \mathcal{K}$, we solve $LP_A(\mathcal{B}_1, w_3, z_1)$ to determine whether or not the exchange pivot involving w_3 and z_1 will yield an adjacent invariancy region. In order to set up $LP_A(\mathcal{B}_1, w_3, z_1)$ (4.28), we must first compute $\mathcal{Y}^{\xi}_{\mathcal{B}_1}(\theta)$ for each $\xi \in \{w_2, w_4, z_5\}$. Using (4.29) we compute: (i) $\mathcal{Y}^{w_2}_{\mathcal{B}_1}(\theta) = 3\theta_2$, (ii) $\mathcal{Y}^{w_4}_{\mathcal{B}_1}(\theta) = -2\theta_1 - \frac{1}{2}\theta_2 + 8$, and (iii) $\mathcal{Y}^{z_5}_{\mathcal{B}_1}(\theta) = -\frac{1}{2}\theta_2 + 4$. Notice from (4.31) that $w_2 \in (\mathrm{H}^{w_3}_{\mathcal{B}_1})'$. Thus, $LP_A(\mathcal{B}_1, w_3, z_1)$ (4.28) is:

$$\begin{array}{rc} \max\limits_{\lambda,\theta} & \lambda \\ \text{s.t.} & -\frac{1}{2}\theta_1 + 2\theta_2 - \lambda \geq 0 \\ & -\theta_1 - 5\theta_2 - \lambda \geq -8 \\ & -\theta_2 - \lambda \geq -4 \\ & \frac{1}{2}\theta_2 = 0 \\ & -2\theta_1 - \frac{1}{2}\theta_2 - \lambda \geq -8 \\ & -\frac{1}{2}\theta_2 - \lambda \geq -4 \\ & -4 \leq \theta_1 \leq 3 \\ & -2 \leq \theta_2 \leq 2 \end{array}$$

The solution of this LP is $(\theta_1^*, \theta_2^*, \lambda^*) = (-4, 0, 2)$. Since the optimal value is positive, the exchange pivot involving w_3 and z_1 will yield an adjacent invariancy region. Thus, we let $\mathcal{B}_3 = (\mathcal{B}_1 \setminus \{z_1, w_3\}) \cup \{w_1, z_3\}$, $\mathcal{R} = \mathcal{R} \cup \mathcal{IR}_{\mathcal{B}_3}$, $\mathcal{K} = \mathcal{K} \cup \mathcal{B}_3$ and $\mathcal{F}_{\mathcal{B}_3} = \mathcal{F}_{\mathcal{B}_3} \cup \{w_3\}$. The tableau for $\mathcal{B}_3$ is:

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
z_3	0	0	-2	0	-1	1	4	1	1	0	$-\frac{1}{2}\theta_1 + \theta_2$
w_2	0	1	-4	0	0	0	0	0	2	0	$3\theta_2$
w_1	1	0	-1	0	0	0	0	0	-2	0	$-\frac{1}{2}\theta_2$
w_4	0	0	1	1	0	2	-2	0	-4	0	$-2\theta_1 - \frac{1}{2}\theta_2 + 8$
z_5	0	0	1	0	0	0	0	0	0	1	$-\frac{1}{2}\theta_2 + 4$

(4.63)

Recognize from (4.63) and (4.48) that $\mathcal{L}^{w_3}_{\mathcal{B}_1,\mathcal{B}_3} = \{w_2\}$. Thus, we set $\mathcal{L}_{\mathcal{B}_3} = \{w_1, w_2\}$.

w_4 - Since $(T_{\mathcal{B}_1})_{w_4,z_4} \neq 0$, a diagonal pivot can be made. Since $(\mathcal{B}_1 \setminus \{w_4\}) \cup \{z_4\} \notin \mathcal{K}$, we let $\mathcal{B}_4 = (\mathcal{B}_1 \setminus \{w_4\}) \cup \{z_4\}$, $\mathcal{R} = \mathcal{R} \cup \mathcal{IR}_{\mathcal{B}_4}$, $\mathcal{K} = \mathcal{K} \cup \mathcal{B}_4$, $\mathcal{F}_{\mathcal{B}_4} = \mathcal{F}_{\mathcal{B}_4} \cup \{z_4\}$ and $\mathbb{P}_{\mathcal{B}_4} = \mathbb{P}_{\mathcal{B}_4} \cup \{z_4\}$. The tableau for $\mathcal{B}_4$ is:

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
z_1	$-\frac{7}{16}$	0	0	$\frac{5}{16}$	$-\frac{3}{8}$	1	$\frac{7}{8}$	$\frac{3}{8}$	0	0	$-\frac{13}{16}\theta_1 + \frac{7}{16}\theta_2 + \frac{5}{2}$
w_2	$-\frac{7}{8}$	1	0	$\frac{5}{8}$	$\frac{5}{4}$	0	$-\frac{25}{4}$	$-\frac{5}{4}$	0	0	$-\frac{5}{8}\theta_1 + \frac{15}{8}\theta_2 + 5$
w_3	$-\frac{3}{8}$	0	1	$\frac{1}{8}$	$\frac{1}{4}$	0	$-\frac{5}{4}$	$-\frac{1}{4}$	0	0	$-\frac{1}{8}\theta_1 - \frac{1}{8}\theta_2 + 1$
z_4	$-\frac{5}{16}$	0	0	$-\frac{1}{16}$	$-\frac{1}{8}$	0	$\frac{5}{8}$	$\frac{1}{8}$	1	0	$\frac{1}{16}\theta_1 + \frac{5}{16}\theta_2 - \frac{1}{2}$
z_5	$\frac{3}{8}$	0	0	$-\frac{1}{8}$	$-\frac{1}{4}$	0	$\frac{5}{4}$	$\frac{1}{4}$	0	1	$\frac{1}{8}\theta_1 - \frac{3}{8}\theta_2 + 3$

(4.64)

Recognize from (4.64) that $\mathcal{L}^{w_4}_{\mathcal{B}_1,\mathcal{B}_4} = \emptyset$. Thus, we set $\mathcal{L}_{\mathcal{B}_4} = \emptyset$.

This completes iteration 2.

Iteration 3: (Here $\mathcal{IR}_{\mathcal{B}_2}$ is considered. The details are omitted as they are analogous to those found in Iteration 2. A new basis $\mathcal{B}_5 = \{z_1, z_2, w_3, z_4, z_5\}$ is discovered.)

Iteration 4: Remove $\mathcal{IR}_{\mathcal{B}_3}$ from $\mathcal{R}$, resulting in $\mathcal{R} = \{\mathcal{IR}_{\mathcal{B}_4}, \mathcal{IR}_{\mathcal{B}_5}\}$. The tableau for $\mathcal{B}_3$ is shown in (4.63). Since $\mathcal{L}_{\mathcal{B}_3} \neq \emptyset$ we consider each of its elements.

w_1 - Notice from (4.63) that $(T_{\mathcal{B}_3})_{w_1,z_1} = 0$ and therefore a diagonal pivot cannot be made on this element. Exchange pivots involving z_3 and w_4 are possible, though. Since $(\mathcal{B}_3 \setminus \{w_1, z_3\}) \cup \{z_1, w_3\} = \mathcal{B}_1$ we do not perform this pivot. However, $(\mathcal{B}_3 \setminus \{w_1, w_4\}) \cup \{z_1, z_4\} \notin \mathcal{K}$ so we solve $LP_{A2}(\mathcal{B}_3, w_1, w_4)$ (4.49). The solution of this LP is $(\theta_1^*, \theta_2^*, \lambda^*) = (3, 0, -2.5)$. Since the optimal value is negative, the exchange pivot involving w_1 and w_4 will not yield an adjacent invariancy region.

w_2 - Since $(T_{\mathcal{B}_3})_{w_2,z_2} = 0$, a diagonal pivot cannot be made. The only exchange pivot that is possible involves z_3. Since $(\mathcal{B}_3 \setminus \{w_2, z_3\}) \cup \{z_2, w_3\} = \mathcal{B}_2$ we do not perform this pivot.

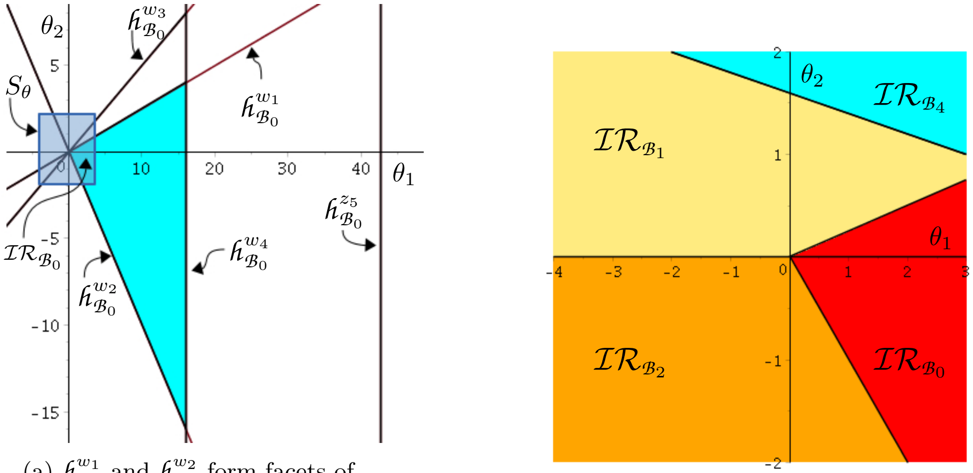
This completes iteration 4.

Iterations 5 and 6: (In these iterations $\mathcal{IR}_{\mathcal{B}_4}$ and $\mathcal{IR}_{\mathcal{B}_5}$ are considered. The details are omitted since they are analogous to work previously done, and no new feasible complementary bases are obtained.)

At the end of iteration 6 we have $\mathcal{R} = \emptyset$, and thus the algorithm terminates, having revealed full-dimensional invariancy regions:

- $\mathcal{IR}_{\mathcal{B}_0} = \{\theta \in S_\theta : \theta_1 - 4\theta_2 \geq 0, \theta_1 + \theta_2 \geq 0\}$
- $\mathcal{IR}_{\mathcal{B}_1} = \{\theta \in S_\theta : -\theta_1 + 4\theta_2 \geq 0, \theta_2 \geq 0, -\theta_1 - 5\theta_2 + 8 \geq 0\}$
- $\mathcal{IR}_{\mathcal{B}_2} = \{\theta \in S_\theta : -\theta_2 \geq 0, -\theta_1 - \theta_2 \geq 0\}$
- $\mathcal{IR}_{\mathcal{B}_4} = \{\theta \in S_\theta : \theta_1 + 5\theta_2 - 8 \geq 0\}$

To see that these invariancy regions do in fact partition S_θ, observe Figure 4.7b.



(a) $\hbar^{w_1}_{\mathcal{B}_0}$ and $\hbar^{w_2}_{\mathcal{B}_0}$ form facets of $\mathcal{IR}_{\mathcal{B}_0}$

(b) Partition of S_θ

Figure 4.7: Example Visual Aids

Notice that this partition of S_θ provides the following solution to the mpLCP:

$\theta \in$	$\mathcal{IR}_{\mathcal{B}_0}$	$\mathcal{IR}_{\mathcal{B}_1}$	$\mathcal{IR}_{\mathcal{B}_2}$	$\mathcal{IR}_{\mathcal{B}_4}$
$w =$	$\begin{bmatrix} \frac{1}{4}\theta_1 - \theta_2 \\ \theta_1 + \theta_2 \\ \frac{1}{4}\theta_1 - \frac{1}{2}\theta_2 \\ -\frac{9}{48}\theta_1 + 8 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 5\theta_2 \\ \frac{1}{2}\theta_2 \\ -\theta_1 - 5\theta_2 + 8 \\ 0 \end{bmatrix}$	$\begin{bmatrix} -\frac{5}{4}\theta_1 \\ 0 \\ -\frac{3}{4}\theta_2 \\ 0 \\ -\frac{9}{4}\theta_1 + 8 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -\frac{5}{8}\theta_1 + \frac{15}{8}\theta_2 + 5 \\ -\frac{1}{8}\theta_1 - \frac{1}{8}\theta_2 + 1 \\ 0 \\ 0 \end{bmatrix}$
$z =$	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\frac{1}{4}\theta_1 + 4 \end{bmatrix}$	$\begin{bmatrix} -\frac{1}{2}\theta_1 + 2\theta_2 \\ 0 \\ 0 \\ 0 \\ -\theta_2 + 4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -\frac{1}{8}\theta_1 - \frac{1}{8}\theta_2 \\ 0 \\ 0 \\ \frac{1}{4}\theta_2 + 4 \end{bmatrix}$	$\begin{bmatrix} -\frac{13}{16}\theta_1 + \frac{7}{16}\theta_2 + \frac{5}{2} \\ 0 \\ 0 \\ \frac{1}{16}\theta_1 + \frac{5}{16}\theta_2 - \frac{1}{2} \\ \frac{1}{8}\theta_1 - \frac{3}{8}\theta_2 + 3 \end{bmatrix}$

4.B Appendix B: Complete Experimental Results

Table 4.2: Complete Results for $k = 2$ – MPT results in standard font; proposed method in bold.

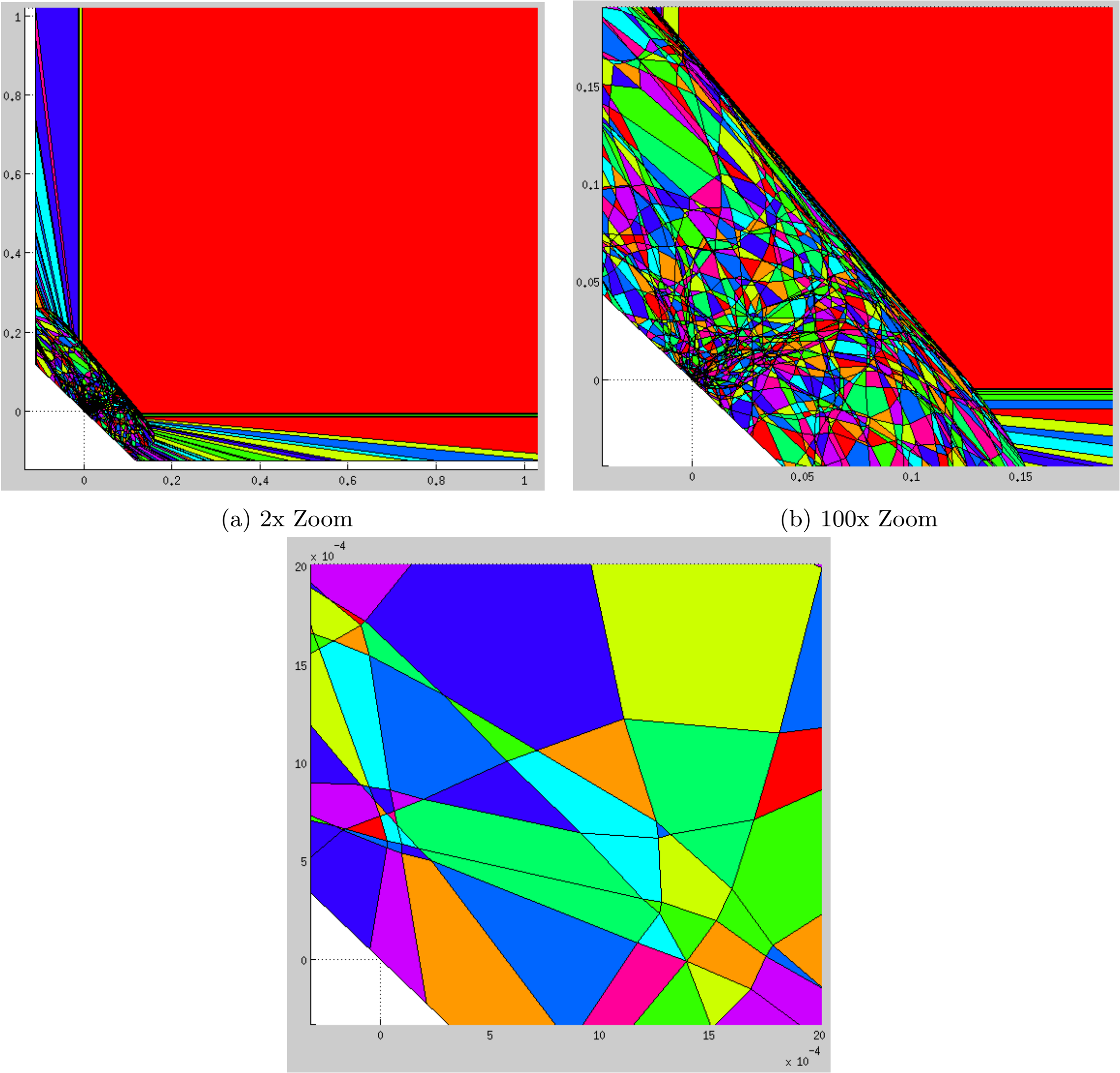
Instance	h	Total time (s)		Number of Regions		Ph1 Iter.	Ph2 Iter.
TAME	4	1.17	**0.03**	5	**5**	**1**	**6**
HS35	4	1.38	**0.03**	10	**10**	**2**	**10**
QPTEST	5	1.48	**0.03**	7	**7**	**1**	**7**
HS35mod	5	1.91	**0.04**	14	**14**	**2**	**14**
ZECEVIC2	6	1.16	**0.02**	6	**6**	**1**	**6**
HS21	7	2.91	**0.03**	13	**13**	**1**	**13**
HS76	7	4.45	**0.03**	26	**26**	**1**	**26**
HS268	15	0.16	**0.04**	‡	**18**	**4**	**18**
S268	15	0.13	**0.03**	‡	**15**	**4**	**15**
HS51	16	6.68	**0.05**	35	**35**	**1**	**43**
HS52	16	6.46	**0.05**	34	**34**	**1**	**40**
HS53	26	8.58	**0.08**	45	**45**	**2**	**45**
LOTSCHD	26	2.41	**0.05**	12	**12**	**18**	**12**
GENHS28	36	24.04	**0.34**	91	**167**	**1**	**177**
QAFIRO	67	0.17	**2.18**	‡	**436**	**2**	**519**
Qplib_3550	105	140.97	**0.37**	411	**37**	**1**	**83**
DUAL4	152	fail	**fail**	–	–	–	–
QADLITTL	168	250.83	**fail**	961	–	–	–
DUAL1	172	898.64	**87.03**	4191	**4165**	**1**	**4169**
QSHARE2B	188	fail	**2652.22**	–	**62924**	**3**	**74671**
DUAL2	194	990.68	**100.70**	4349	**4292**	**1**	**4293**
QPCBLEND	200	fail	**fail**	–	–	–	–
DUAL3	224	1746.10	**200.83**	6348	**6223**	**1**	**6223**
DUALC1	234	73.62	**0.78**	30	**31**	**1**	**33**
PRIMALC2	240	125.13	**1.34**	61	**50**	**1**	**151**
DUALC2	244	93.21	**0.72**	32	**29**	**1**	**35**
CVXQP2S	250	275.18	**35.31**	1023	**1023**	**9**	**1097**
PRIMALC1	254	439.99	**3.99**	142	**128**	**1**	**191**
DUALC5	295	62.39	**0.86**	24	**24**	**1**	**26**
CVXQP1S	300	1085.30	**221.00**	4177	**4667**	**6**	**4729**
PRIMALC5	304	401.77	**6.90**	179	**168**	**1**	**226**
QISRAEL	316	0.57	**†**	‡	**33743**	**164**	**48955**
QSCAGR7	353	22.14	**41.30**	16	**405**	**1061**	**425**
Qplib_2883	374	fail	**140.38**	–	**1631**	**27**	**2398**
DPKLO1	420	fail	**†**	–	**31910**	**10**	**37076**
QSHARE1B	431	70.73	**1593.50**	62	**12122**	**588**	**13265**
QRECIPE	433	2.76	**106.22**	‡	**1186**	**3**	**1908**
QSC205	499	3479.40	**139.96**	1930	**1186**	**1**	**2328**
DUALC8	520	262.43	**fail**	56	–	–	–
QE226	538	†	**185.19**	1583	**1249**	**4**	**3012**
PRIMALC8	545	248.88	**18.85**	212	**212**	**1**	**222**
Qplib_3965	566	†	**†**	10075	**23655**	**9**	**23655**
Qplib_3970	566	†	**3389.96**	9349	**21126**	**51**	**21126**
Qplib_3975	566	fail	**†**	–	**23398**	**39**	**23429**
Qplib_3977	566	fail	**3110.91**	–	**21025**	**49**	**21025**
Qplib_3839	574	fail	**125.67**	–	**867**	**1**	**1894**
QBEACONF	575	1119.80	**45.58**	311	**318**	**99**	**380**
QBRANDY	635	fail	**758.77**	–	**2046**	**43**	**3681**
Qplib_3897	722	fail	**†**	–	**16451**	**80**	**16451**
Qplib_3907	722	fail	**†**	–	**15600**	**67**	**15600**
Qplib_3908	722	†	**†**	6048	**14736**	**74**	**14736**
PRIMAL1	734	fail	**†**	–	**13234**	**1**	**15123**
QGROW7	861	24.85	**†**	‡	**5298**	**3**	**7535**
QSCTAP1	900	†	**2083.78**	189	**4373**	**38**	**5944**
QSCFXM1	974	1854.10	**†**	116	**5624**	**279**	**6712**
QSCORPIO	1026	†	**†**	175	**7696**	**1**	**9796**
KSIP	1041	2394.80	**524.69**	1238	**1322**	**7**	**1347**
QBANDM	1082	†	**931.61**	282	**1547**	**53**	**2896**
Qplib_2461	1269	†	**†**	133	**3959**	**147**	**4999**
PRIMAL2	1393	†	**†**	41	**4717**	**1**	**5690**

† – Time limit of one hour reached. ‡ – Reported as infeasible.

Table 4.3: Complete Results for $k = 3$ – MPT results in standard font; proposed method in bold.

Instance	k	Total time (s)		Number of Regions		Ph1 Iter.	Ph2 Iter.
TAME	4	2.02	**0.03**	9	**9**	**1**	**9**
HS35	4	2.86	**0.03**	15	**15**	**1**	**15**
QPTEST	5	2.75	**0.03**	12	**12**	**1**	**12**
HS35mod	5	3.23	**0.03**	18	**18**	**1**	**18**
ZECEVIC2	6	3.45	**0.03**	13	**13**	**1**	**13**
HS21	7	5.40	**0.03**	24	**24**	**1**	**24**
HS76	7	12.70	**0.04**	62	**62**	**1**	**62**
HS268	15	0.52	**0.05**	1	**27**	**4**	**27**
S268	15	7.46	**0.05**	23	**23**	**8**	**23**
HS51	16	32.49	**0.12**	111	**111**	**1**	**111**
HS52	16	45.49	**0.18**	169	**169**	**1**	**169**
HS53	26	166.59	**1.31**	423	**424**	**1**	**424**
LOTSCHD	26	18.63	**0.21**	52	**52**	**12**	**52**
GENHS28	36	384.25	**2.32**	663	**666**	**1**	**667**
QAFIRO	67	352.43	**53.04**	501	**5994**	**3**	**6110**
Qplib_3550	105	†	**117.21**	5631	**6503**	**1**	**6618**
DUAL4	152	fail	**fail**	–	–	–	–
QADLITTL	168	fail	**fail**	–	–	–	–
DUAL1	172	†	**2744.41**	10862	**106593**	**1**	**106593**
QSHARE2B	188	fail	†	–	**19392**	**9**	**27447**
DUAL2	194	fail	†	–	**111742**	**1**	**111742**
QPCBLEND	200	†	**fail**	9106	–	–	–
DUAL3	224	fail	†	–	**85731**	**1**	**85731**
DUALC1	234	404.57	**4.11**	121	**121**	**1**	**124**
PRIMALC2	240	1753.70	**29.78**	1546	**959**	**1**	**1035**
DUALC2	244	303.93	**5.75**	98	**172**	**1**	**209**
CVXQP2S	250	fail	**2727.38**	–	**53810**	**1**	**53810**
PRIMALC1	254	fail	**60.58**	–	**1911**	**1**	**1989**
DUALC5	295	170.96	**3.41**	75	**76**	**1**	**79**
CVXQP1S	300	fail	†	–	**58436**	**4**	**58477**
PRIMALC5	304	1286.20	**41.29**	932	**841**	**1**	**1399**
QISRAEL	316	0.69	†	‡	**20271**	**109**	**22873**
QSCAGR7	353	fail	**469.20**	–	**5149**	**888**	**5353**
Qplib_2883	374	fail	†	–	**7082**	**15**	**9133**
DPKLO1	420	†	†	6218	**13365**	**7**	**19997**
QSHARE1B	431	†	**38.52**	1379	**101**	**442**	**103**
QRECIPE	433	3.97	†	‡	**20457**	**2**	**21259**
QSC205	499	fail	†	–	**16401**	**1**	**24438**
DUALC8	520	fail	**fail**	–	–	–	–
QE226	538	†	†	423	**9145**	**7**	**12419**
PRIMALC8	545	fail	**605.36**	–	**2754**	**1**	**2812**
Qplib_3965	566	fail	†	–	**12683**	**2**	**12683**
Qplib_3970	566	†	†	3681	**13177**	**47**	**13177**
Qplib_3975	566	fail	†	–	**13538**	**11**	**13538**
Qplib_3977	566	†	†	4629	**13588**	**1**	**13588**
Qplib_3839	574	fail	†	–	**14205**	**1**	**14745**
QBEACONF	575	†	**1613.43**	616	**5913**	**127**	**8599**
QBRANDY	635	†	**2566.00**	601	**738**	**16**	**2652**
Qplib_3897	722	†	†	3417	**8805**	**1**	**8902**
Qplib_3907	722	†	†	3624	**9212**	**1**	**9212**
Qplib_3908	722	†	†	2533	**8243**	**49**	**8243**
PRIMAL1	734	†	†	1788	**8018**	**1**	**9040**
QGROW7	861	†	†	96	**3566**	**1**	**4048**
QSCTAP1	900	†	†	96	**4610**	**12**	**4843**
QSCFXM1	974	610.12	†	78	**3787**	**228**	**3794**
QSCORPIO	1026	†	†	95	**4146**	**1**	**5246**
KSIP	1041	†	†	555	**4111**	**2**	**4136**
QBANDM	1082	†	†	91	**2970**	**63**	**5592**
Qplib_2461	1269	†	†	54	**2748**	**4**	**2868**
PRIMAL2	1393	†	†	179	**2526**	**1**	**2851**

† – Time limit of one hour reached. ‡ – Reported as infeasible.



(a) 2x Zoom

(b) 100x Zoom

(c) 1000x Zoom

Figure 4.8: Three scaled images of invariancy regions for instance DUAL1 with $k = 2$.

Chapter 5

A Two-phase Algorithm for mpLCP with Parameters in General Locations

[The contents of this chapter include material from a paper entitled “On the multiparametric linear complementarity problem with parameters in general locations,” which will be submitted to the journal *Mathematics of Operations Research* in August of 2016; the author is N. Adelgren.]

5.1 Introduction

In this work we consider the multiparametric form of the Linear Complementarity Problem (LCP) in which all input data is permitted to be dependent on a vector of parameters $\theta \in S_\theta \subseteq \mathbb{R}^k$, where S_θ is a bounded convex polytope defining the set of “attainable” values for θ. This problem is referred to as the multiparametric Linear Complementarity Problem (mpLCP). Let $\Theta = \{\alpha^\top \theta + \beta : \alpha \in \mathbb{R}^k, \beta \in \mathbb{R}\}$, the set of affine functions of θ. Then mpLCP is as follows:

> Given $M(\theta) \in \Theta^{h \times h}$, $q(\theta) \in \Theta^h$, for each $\theta \in S_\theta$ find vectors $w(\theta)$ and $z(\theta)$ that satisfy the following system:
>
> $$\begin{aligned} w - M(\theta)z &= q(\theta) \\ w^\top z &= 0 \\ w, z &\geq 0 \end{aligned} \tag{5.1}$$

If such a solution exists for a given $\theta \in S_\theta$, mpLCP is said to be *feasible at* θ, otherwise it is *infeasible at* θ. Similarly, mpLCP is said to be *feasible* if there exists a $\hat{\theta} \in S_\theta$ at which mpLCP is feasible, otherwise mpLCP is *infeasible.* As finding a solution to (5.1) for each $\theta \in S_\theta$ individually is intractable, the goal of mpLCP is to partition the space S_θ into regions such that the representation of the solution vectors w and z as functions of θ is invariant over each region. In the literature these regions have been given a variety of names, such as invariancy regions, critical regions, and validity sets. We refer to them as *invariancy regions.* A detailed discussion on invariancy regions and their properties is provided in Section 5.2.

Note that the nonparametric version of LCP has the same form as (5.1) with the exception that $M(\theta)$ and $q(\theta)$ are replaced by $M \in \mathbb{R}^{h \times h}$ and $q \in \mathbb{R}^h$, respectively. LCP is a well known problem in the literature and has been studied extensively by researchers such as Kostreva [55], Lemke [56], Murty and Yu [65] and Cottle et al. [22]. Though LCP is NP-hard in general, polynomial time algorithms exist for certain classes of the matrix M. Thus, much work has been done in order to identify various classes of matrices M which impact one's ability to solve an instance of LCP. Solution techniques for LCP are often designed for specific classes of M. For a concise list of important matrix classes see [21]. For a detailed discussion on these classes and their impact on LCP see [22, 65]. We will refer to many of the matrix classes discussed in these works throughout this chapter.

Single parametric LCP (pLCP) with a parameter present only in the $q(\theta)$ vector (i.e., $k = 1$ and $M(\theta) = M \in \mathbb{R}^{h \times h}$) was first proposed as a result of the work done by Maier [58] and has been studied extensively since. Columbano et al. [18], Gailly et al. [32], Li and Ierapetritou [57] and Adelgren and Wiecek [2] consider (5.1) with $k > 1$ and $M(\theta) = M \in \mathbb{R}^{h \times h}$. The method of Gailly et al. [32] is designed for the case in which M is a copositive-plus matrix, the methods of Columbano et al. [18] and Adelgren and Wiecek [2] are designed for instances in which M is a sufficient matrix, and the method of Li and Ierapetritou [57] works for general M, but is computationally expensive since it requires reformulating the mpLCP as a multiparametric bilinear mixed integer program. Parametric and multiparametric LCP in which $M(\theta) \neq M \in \mathbb{R}^{h \times h}$ (i.e., the matrix $M(\theta)$ cannot be represented as a real valued matrix, as it depends on θ) has received little attention though. Interesting properties of the case in which $M(\theta) \neq M \in \mathbb{R}^{h \times h}$ are discussed by Tammer [93]. Xiao [103] and Chakraborty et al. [17] present solution techniques for the case when $M(\theta) \neq M \in \mathbb{R}^{h \times h}$, but restrict that $M(\theta)$ be a P-matrix for all $\theta \in S_\theta$.

The method we propose in this chapter extends the works of Väliaho [98], who considered (5.1) for $k = 1$, and Adelgren and Wiecek [2], who considered (5.1) for $M(\theta) = M \in \mathbb{R}^{h \times h}$ and $k > 1$, and solves (5.1) whenever it is feasible and $M(\theta)$ is a sufficient matrix for each $\theta \in S_\theta$. Since we require that $M(\theta)$ be sufficient, we provide the following definition, as found in [22].

Definition 5.1. A matrix $M \in \mathbb{R}^{h \times h}$ is *column sufficient* if the following implication is satisfied:

$$(x_i(Mx)_i \leq 0 \text{ for all } i) \quad \Rightarrow \quad (x_i(Mx)_i = 0 \text{ for all } i) \tag{5.2}$$

M is said to be *row sufficient* if $M^\top$ is column sufficient. If M is both column and row sufficient, it is then called *sufficient*.

Assumption 5.2. We assume that $M(\theta)$ is a sufficient matrix for all $\theta \in S_\theta$.

We note that although there exist finite time algorithms capable of determining whether or not a given matrix is sufficient (see, for example, [100]), in general, determining whether or not $M(\theta)$ is sufficient for all $\theta \in S_\theta$ is not a trivial task. To see this, consider the following results.

Lemma 5.3. (Theorem 4.6 of Väliaho [99]) The matrix $A = \begin{bmatrix} 0 & B \\ C & 0 \end{bmatrix} \in \mathbb{R}^{n \times n}$ where the zero blocks are square, is sufficient if and only if all the corresponding minors of B and $-C^\top$ have the same sign.

Proposition 5.4. The set of sufficient matrices is not closed under convex combinations.

Proof. Consider the matrices $A^1 = \begin{bmatrix} 0 & 2 \\ -1 & 0 \end{bmatrix}$ and $A^2 = \begin{bmatrix} 0 & -1 \\ 2 & 0 \end{bmatrix}$. It is easy to see that both A^1 and A^2 satisfy the conditions of Lemma 5.3 and are therefore sufficient. For each $\alpha \in [0, 1]$ consider the following

$$\begin{aligned} \alpha A^1 + (1-\alpha)A^2 &= \begin{bmatrix} 0 & 2\alpha \\ -\alpha & 0 \end{bmatrix} + \begin{bmatrix} 0 & \alpha - 1 \\ 2 - 2\alpha & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 3\alpha - 1 \\ 2 - 3\alpha & 0 \end{bmatrix} \end{aligned}$$

It is clear that for all $\alpha \in (\frac{1}{3}, \frac{2}{3})$ the above matrix does not satisfy the conditions of Lemma 5.3. Thus, the above matrix is not sufficient for any $\alpha \in (\frac{1}{3}, \frac{2}{3})$. This clearly shows that the set of sufficient matrices is not closed under convex combinations. □

The result of Proposition 5.4 also shows that the set of sufficient matrices cannot be closed under conic or linear combinations. Thus, even in the case in which S_θ is translated into the nonnegative orthant and there exist sufficient matrices $M^0, M^1, \ldots, M^k \in \mathbb{R}^{h\times h}$ such that $M(\theta) = M^0 + \sum_{i=1}^{k} M^i\theta_i$, one is not guaranteed that $M(\theta)$ is sufficient for all $\theta \in S_\theta$. This serves as evidence that the problem of determining whether or not $M(\theta)$ is sufficient for all $\theta \in S_\theta$ is not trivial, in general. Although we acknowledge the difficulty of this problem, solving it is not the focus of this work. We note that, in practice, satisfaction of the stronger condition that $M(\theta)$ be positive semi-definite for all $\theta \in S_\theta$ is easier to verify. Even under this condition, though, the procedures we present still allow one to solve several important parametric problems for which there was previously no available solution technique. We discuss a variety of such problems in the next several paragraphs, and we note that for each of these problems $M(\theta)$ will always be positive semi-definite for all $\theta \in S_\theta$.

Since mpLCP as in (5.1) has not yet been studied for $k > 1$, we briefly discuss some of the problems this method allows one to solve. It is well known that both linear programs (LPs) and convex quadratic programs (QPs) can be reformulated as LCPs. Thus, (5.1) encompasses two very important classes of problems:

(i) Multiparametric LP (mpLP):

$$\begin{array}{rc} \min\limits_{x} & c(\theta)^\top x \\ \text{s.t.} & A(\theta)x \leq b(\theta) \end{array} \tag{5.3}$$

(ii) Multiparametric (convex) QP (mpQP):

$$\begin{array}{rc} \min\limits_{x} & \frac{1}{2}x^\top Q(\theta)x + c(\theta)^\top x \\ \text{s.t.} & A(\theta)x \leq b(\theta) \end{array} \tag{5.4}$$

Parametric LP with $A(\theta) = A$ and parametric QP with $A(\theta) = A$ and $Q(\theta) = Q$ have been studied extensively. Pistikopoulos et al. [75] provide an excellent survey of the literature for the case in which $k > 1$. Parametric LP with $A(\theta) \neq A$ has also been studied for quite some time. Perhaps the earliest work is due to Courtillot [23]. Solution techniques for various special cases of parametric LP with $k = 1$ and $A(\theta) \neq A$ are presented in works such as Barnett [6], Dent et al. [26], Finkelstein and Gumenok [31], Kim [52], Willner [102] and Filar et al. [30]. The works of Väliaho [96] and Khalilpour and Karimi [51] introduce methods for solving (5.3) with $k = 1$ and $A(\theta) \neq A$. We are unaware of any work that provides a method for solving (5.3) with $k > 1$ and $A(\theta) \neq A$. However, the method we present is capable of solving this problem.

In addition to the works cited in Pistikopoulos et al. [75], parametric QP with $A(\theta) = A$ and $Q(\theta) = Q$ is considered in Ghaffari-Hadigheh et al. [34] and Ghaffari-Hadigheh et al. [35], particularly for the case in which $k = 2$. Solution techniques are presented for (5.4) with $k = 1$, $A(\theta) \neq A$ and $Q(\theta) \neq Q$ in Ritter [81], Väliaho [97] and Jonker et al. [47]. As with mpLP, we are unaware of any work that provides a method for solving (5.4) with $k > 1$, $A(\theta) \neq A$ and $Q(\theta) \neq Q$. Again, the method we present is capable of solving this problem.

Another class of problems that can be reformulated and solved using (5.1) is multiobjective programming problems having linear and/or convex quadratic objective functions and linear constraints. This is due to the common method of solving multiobjective problems using scalarization techniques which transform the problem into a single objective problem by introducing one or more parameters. For a detailed discussion on multiobjective programming and the various scalarization techniques available, see Ehrgott [27]. Multiobjective programs in which all objective functions are linear have been widely studied and can be solved efficiently using the multiobjective simplex method (Ehrgott [27]). Efficient methods have also been proposed for problems with one convex quadratic objective and one or more linear objectives, see, for example Hirschberger et al. [44], Steuer et al. [89] and Hirschberger et al. [43]. Goh and Yang [36] present a method for solving multiobjective problems with two or more convex quadratic objectives, though they impose a few minor restrictions. The work we present here serves as an alternative method for solving multiobjective programs with any number of linear and/or convex quadratic objectives, without restriction.

The method for solving mpLCP (5.1) which we present in this work is a two-phase method. We will show that the problem solved in the first phase of this method is simply a special case of the problem solved during the second phase. For this reason we discuss Phase 2 prior to Phase 1. Hence, the remainder of this chapter is organized as follows. Background information on LCP problems and their geometrical structure is contained in Section 5.2. In Section 5.3 we discuss algebraic properties of invariancy regions. The theory and methodology for Phase 2 of the proposed method for solving mpLCP are presented in Section 5.4. In Section 5.5 we present the theory and methodology for Phase 1. We discuss the uniqueness of partitions of S_θ in Section 5.6. In Section 5.7 we present the results of an experiment and discuss the performance of the proposed algorithms. Finally, in Section 5.8 we provide concluding remarks and a discussion on proposed future work. We also include two appendices in Sections 5.A and 5.B in which we provide tables that are important for the examples we discuss in Sections 5.4–5.6.

5.2 Background on mpLCP

This section is divided into four subsections. In the first we present preliminary notations and definitions. In the second we provide a detailed discussion on invariancy regions and the properties of these sets. In the third subsection we discuss the geometry of mpLCP and provide some preliminary results. Finally, in the last subsection we delve deeper into the details of invariancy regions and develop the theory which shows that S_θ can, in fact, be partitioned. As many of the concepts we introduce throughout this work are new and rather difficult, we introduce here two examples of small instances of mpLCP (5.1) which we will refer back to at various locations throughout this chapter.

Example 5.5.

$$\begin{gathered} w - \begin{bmatrix} 0 & 0 & 1 & 3 & -5 \\ 0 & 0 & 2 & 2 & 2 \\ -1 & -2 & 2\theta_1 - \theta_2 + 4 & \theta_1 - 2\theta_2 + 3 & 3\theta_1 + 4\theta_2 - 2 \\ -3 & -2 & \theta_1 - 2\theta_2 + 3 & -\theta_1 + \theta_2 + 4 & 3\theta_1 + 4\theta_2 - 3 \\ 5 & -2 & 3\theta_1 + 4\theta_2 - 2 & 3\theta_1 + 4\theta_2 - 3 & -\theta_2 + 3 \end{bmatrix} z = \begin{bmatrix} 3 \\ -\theta_1 - 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ w^\top z = 0 \\ w, z \geq 0 \end{gathered} \tag{5.5}$$

Notice here that $h = 5$ and $k = 2$. Assume that $S_\theta = \{\theta \in \mathbb{R}^2 : \theta \geq 0, \theta_1 + \theta_2 \leq 1\}$. It is easy to verify for this example that $M(\theta)$ is sufficient for all $\theta \in S_\theta$ since $M(\theta)$ is actually positive semi-definite for each $\theta \in S_\theta$.

Example 5.6.

$$\begin{gathered} w - \begin{bmatrix} 0 & 0 & -2 & -1 \\ 0 & 0 & -5 & \theta_1 + 7 \\ 1 & 3 & 0 & 0 \\ 1 & -\theta_1 - 5 & 0 & 0 \end{bmatrix} z = \begin{bmatrix} -\theta_2 - 1 \\ \theta_1 - \theta_2 - 1 \\ -18\theta_2 - 34 \\ -9\theta_2 - 17 \end{bmatrix} \\ w^\top z = 0 \\ w, z \geq 0 \end{gathered} \tag{5.6}$$

For this example $h = 4$ and $k = 2$. Assume that $S_\theta = [-3, 1]^2$. It can be verified that $M(\theta)$ is sufficient for all $\theta \in S_\theta$ by using Lemma 5.3.

5.2.1 Preliminaries

We begin this subsection by introducing definitions and notation necessary for the remainder of this work. Many of these definitions slight modifications of traditional definitions for LCP. Assume that we are given an mpLCP of the form (5.1) and define the matrix $G(\theta) := \begin{bmatrix} I & -M(\theta) \end{bmatrix}$ and the vector $\nu := \begin{bmatrix} w \\ z \end{bmatrix}$, where $G(\theta) \in \mathbb{R}^{h \times h} \times \Theta^{h \times h}$ and $\nu \in \mathbb{R}^{2h}$. We use the notation $G(\theta)_{i\bullet}$ to represent the i^{th} row of $G(\theta)$ and $G(\theta)_{\bullet j}$ to represent the j^{th} column of $G(\theta)$. Also, given a set $\mathcal{I} \subseteq \{1, \ldots, h\}$ we use $G(\theta)_{\mathcal{I}\bullet}$ to denote the matrix formed by the rows of $G(\theta)$ indexed by $\mathcal{I}$. Similarly, given a set $\mathcal{J} \subseteq \{1, \ldots, 2h\}$ we use $G(\theta)_{\bullet\mathcal{J}}$ to denote the matrix formed by the columns of $G(\theta)$ indexed by $\mathcal{J}$. Furthermore, given $\mathcal{I} \subseteq \{1, \ldots, h\}$ and $\mathcal{J} \subseteq \{1, \ldots, 2h\}$, we use $G(\theta)_{\mathcal{I}\mathcal{J}}$ to represent the submatrix of $G(\theta)$ consisting of the elements of the rows indexed by $\mathcal{I}$ which are in the columns indexed by $\mathcal{J}$, i.e., $G(\theta)_{\mathcal{I}\mathcal{J}} = (G(\theta)_{\mathcal{I}\bullet})_{\bullet\mathcal{J}}$. Let $\mathcal{E}$ denote the index set $\{1, \ldots, 2h\}$ for (5.1).

Definition 5.7. Given a set $\mathcal{I} \subset \mathcal{E}$, $\theta \in S_\theta$ is called *rank preserving over* $\mathcal{I}$ if $\mathit{rank}\,(G(\theta)_{\bullet\mathcal{I}}) = |\mathcal{I}|$.

We then let

$$\mathcal{P}(\mathcal{I}) := \{\theta \in S_\theta : \mathit{rank}\,(G(\theta)_{\bullet\mathcal{I}}) = |\mathcal{I}|\} \tag{5.7}$$

be the set of all rank preserving $\theta \in S_\theta$ for $\mathcal{I} \subset \mathcal{E}$.

Definition 5.8. A *basis* is a set $\mathcal{B} \subset \mathcal{E}$ such that $|\mathcal{B}| = h$ and $\mathcal{P}(\mathcal{B}) \neq \emptyset$. The set $\mathcal{N} := \mathcal{E} \setminus \mathcal{B}$ is called the *complement* of $\mathcal{B}$.

Definition 5.9. The sets of variables $\nu_{\mathcal{B}} := \{\nu_i : i \in \mathcal{B}\}$ and $\nu_{\mathcal{N}} := \{\nu_i : i \in \mathcal{N}\}$ are referred to as the sets of *basic* and *nonbasic* variables, respectively.

Definition 5.10. Given a basis $\mathcal{B}$, for every $\theta \in \mathcal{P}(\mathcal{B})$, $\nu_{\mathcal{B}}(\theta) = G(\theta)^{-1}_{\bullet\mathcal{B}}q(\theta)$, $\nu_{\mathcal{N}}(\theta) = 0$ is a solution to the linear system

$$G(\theta)\nu = q(\theta).$$

For each $\theta \in \mathcal{P}(\mathcal{B})$, the solution $\big(\nu_{\mathcal{B}}(\theta), \nu_{\mathcal{N}}(\theta)\big)$ is called a *basic solution.*

Definition 5.11. A basis $\mathcal{B}$ is called *complementary* if $|\{i, i+h\} \cap \mathcal{B}| = 1$ for each $i \in \{1, \ldots, h\}$.

We have now built the tools necessary for providing the definition of an *invariancy region.* Although, we do provide the definition here, we note that Section 5.2.2 is devoted to a more detailed discussion on these sets.

Consider a complementary basis $\mathcal{B}$ and suppose there exists $\theta \in \mathcal{P}(\mathcal{B}) \subseteq S_\theta$ such that: (i) $\nu_{\mathcal{B}}(\theta) = G(\theta)^{-1}_{\bullet\mathcal{B}}q(\theta) \geq 0$ and, (ii) $\nu_{\mathcal{N}}(\theta) = 0$. Then since $\nu = \begin{bmatrix} w \\ z \end{bmatrix}$, for all $\theta \in \mathcal{P}(\mathcal{B})$ satisfying (i) and (ii) above, the basic solution $\big(\nu_{\mathcal{B}}(\theta), \nu_{\mathcal{N}}(\theta)\big)$ satisfies (5.1) and therefore defines solution vectors $w(\theta)$ and $z(\theta)$ for mpLCP. Note that one set of solution vectors of this form may exist for each complementary basis. Hence, there may exist one invariancy region for each complementary basis.

Definition 5.12. The *invariancy region* $\mathcal{IR}_{\mathcal{B}}$ of a complementary basis $\mathcal{B}$ is the set:

$$\mathcal{IR}_{\mathcal{B}} := \left\{\theta \in \mathcal{P}(\mathcal{B}) : G(\theta)^{-1}_{\bullet\mathcal{B}}q(\theta) \geq 0\right\} \tag{5.8}$$

Definition 5.13. A complementary basis $\mathcal{B}$ is called *feasible* to (5.1) if $\mathcal{IR}_{\mathcal{B}} \neq \emptyset$.

Every invariancy region is a possibly non-convex subset of S_θ. For every feasible complementary basis $\mathcal{B}$, the function $\nu_{\mathcal{B}}(\theta) = G(\theta)^{-1}_{\bullet\mathcal{B}}q(\theta)$, $\nu_{\mathcal{N}}(\theta) = 0$ is a solution to (5.1) for all $\theta \in \mathcal{IR}_{\mathcal{B}}$. Furthermore, as will be discussed in more detail in Section 5.2.3, when $M(\theta)$ is sufficient for each $\theta \in S_\theta$ there is an onto correspondence between solutions to mpLCP and complementary bases, i.e., for each $\theta \in S_\theta$ there exists a complementary basis $\mathcal{B}$ for which $(\nu_{\mathcal{B}}, \nu_{\mathcal{N}})$ solves mpLCP at θ. Therefore in this work we propose a method for determining a piecewise solution to (5.1) by partitioning S_θ into a set of invariancy regions with disjoint interiors, where each invariancy region is associated with a unique feasible complementary basis.

5.2.2 Invariancy Regions

In order to gain a deeper understanding of invariancy regions, we return to Examples 5.5 and 5.6. For Example 5.5 it is not difficult to verify that for $\theta = \begin{bmatrix} \frac{1}{5} \\ \frac{2}{5} \end{bmatrix}$, the basis $\mathcal{B}^{5.5}_0 = \{w_1, z_2, z_3, z_4, z_5\}$ is feasible (see Table 5.11 in Section 5.A). Similarly, for Example 5.6, the basis $\mathcal{B}^{5.6}_0 = \{w_1, w_2, w_3, w_4\}$ is feasible for $\theta = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$ (see Table 5.14 in Section 5.B). Note that for ease of understanding we are representing $\mathcal{B}^{5.5}_0$ and $\mathcal{B}^{5.6}_0$ as the sets of variables given by the indices in each basis, rather than the sets of indices themselves. We will often represent bases this way throughout this work. Observe the invariancy regions for $\mathcal{B}^{5.5}_0$ and $\mathcal{B}^{5.6}_0$:

$$\mathcal{IR}_{\mathcal{B}^{5.5}_0} = \left\{ \theta \in \mathcal{P}(\mathcal{B}^{5.5}_0) : \begin{array}{c} \dfrac{3\theta_1^3 + 18\theta_1^2\theta_2 - 49\theta_1^2 - 75\theta_1\theta_2^2 + 148\theta_1\theta_2 + 68\theta_1 + 96\theta_2^2 - 16\theta_2 - 76}{2(-3\theta_1^2 + 8\theta_1\theta_2 + 19\theta_1 + 41\theta_2^2 - 24\theta_2 - 22)} \geq 0 \\ \dfrac{-(\theta_1 + 2)(9\theta_1^3 - 9\theta_1^2\theta_2 - 33\theta_1^2 - 87\theta_1\theta_2^2 + 21\theta_1\theta_2 + 22\theta_1 - 59\theta_2^3 + 13\theta_2^2 + 50\theta_2 + 5)}{4(-3\theta_1^2 + 8\theta_1\theta_2 + 19\theta_1 + 41\theta_2^2 - 24\theta_2 - 22)} \geq 0 \\ \dfrac{(\theta_1 + 2)(-6\theta_1^2 - \theta_1\theta_2 + 11\theta_1 + 15\theta_2^2 - 16\theta_2 + 1)}{2(-3\theta_1^2 + 8\theta_1\theta_2 + 19\theta_1 + 41\theta_2^2 - 24\theta_2 - 22)} \geq 0 \\ \dfrac{(\theta_1 + 2)(3\theta_1^2 + 8\theta_1\theta_2 - \theta_1 + 5\theta_2^2 + 5\theta_2 - 11)}{2(-3\theta_1^2 + 8\theta_1\theta_2 + 19\theta_1 + 41\theta_2^2 - 24\theta_2 - 22)} \geq 0 \\ \dfrac{(\theta_1 + 2)(9\theta_1 - 13\theta_2 + \theta_1\theta_2 + 21\theta_2^2 - 12)}{2(-3\theta_1^2 + 8\theta_1\theta_2 + 19\theta_1 + 41\theta_2^2 - 24\theta_2 - 22)} \geq 0 \end{array} \right\}$$

$$\mathcal{IR}_{\mathscr{B}_0^{5.6}} = \left\{ \theta \in \mathcal{P}(\mathscr{B}_0^{5.6}) : \begin{array}{c} -\theta_2 - 1 \geq 0 \\ \theta_1 - \theta_2 - 1 \geq 0 \\ -18\theta_2 - 34 \geq 0 \\ -9\theta_2 - 17 \geq 0 \end{array} \right\}$$

The following propositions arise from conclusions one may draw from observing an invariancy region such as those above.

Proposition 5.14. Given a feasible complementary basis $\mathscr{B}$, the invariancy region $\mathcal{IR}_{\mathscr{B}}$ is defined by a set of rational inequalities, all having the same denominator.

Proof. Recognize that $G(\theta)^{-1}_{\bullet\mathscr{B}} = \dfrac{Adj\,(G(\theta)_{\bullet\mathscr{B}})}{det\,(G(\theta)_{\bullet\mathscr{B}})}$, where $Adj(\cdot)$ and $det(\cdot)$ represent the matrix adjoint and determinant, respectively. Thus, the result follows from (5.8). □

Proposition 5.14 shows that invariancy regions have a relatively nice structure. However, this structure is actually better than it seems. This is shown in the following lemma and the subsequent proposition.

Lemma 5.15. Given a feasible complementary basis $\mathscr{B}$, one of the following must hold:

1. $det\,(G(\theta)_{\bullet\mathscr{B}}) \geq 0$ for all $\theta \in S_\theta$

2. $det\,(G(\theta)_{\bullet\mathscr{B}}) \leq 0$ for all $\theta \in S_\theta$

Proof. Recall that we assume that $M(\theta)$ is sufficient for each $\theta \in S_\theta$. It is well known that sufficient matrices are also $\mathbf{P_0}$ matrices, i.e., their principal minors are all nonnegative (see, for example, Cottle et al. [19] or Väliaho [99]). Hence, recognize that for any $n \in \{1, \dots, h\}$, the n^{th} order principal minors of $-M(\theta)$ are each: (i) nonnegative for all $\theta \in S_\theta$ if n is even, or (ii) nonpositive for all $\theta \in S_\theta$ if n is odd. Notice that there exists $J \subseteq \{0, \dots, k\}$ such that $G(\theta)_{\bullet\mathscr{B}}$ is obtained from $-M(\theta)$ by replacing column $(-M(\theta))_{\bullet j}$ with $I_{\bullet j}$ for each $j \in J$. Thus, if $|J| = h$, $det\,(G(\theta)_{\bullet\mathscr{B}}) = det(I) = 1$ and if $|J| \neq h$, $det\,(G(\theta)_{\bullet\mathscr{B}})$ is equal to a $(h - |J|)^{th}$ order principal minor of $-M(\theta)$. Therefore, if $(h - |J|)$ is even, condition (1) holds, and if $(h - |J|)$ is odd, condition (2) holds. □

Proposition 5.16. For every feasible complementary basis $\mathscr{B}$, the invariancy region $\mathcal{IR}_{\mathscr{B}}$ is defined by a set of polynomial inqualities.

Proof. From Proposition 5.14 we know that all defining inequalities of $\mathcal{IR}_{\mathscr{B}}$ are given by rational functions whose denominators equal $det\,(G(\theta)_{\bullet\mathscr{B}})$. By Lemma 5.15 we know that either: (1)

$det\,(G(\theta)_{\bullet\mathcal{B}}) \geq 0$ for all $\theta \in S_\theta$ or, (2) $det\,(G(\theta)_{\bullet\mathcal{B}}) \leq 0$ for all $\theta \in S_\theta$. Therefore, since for all $\theta \in \mathcal{P}(\mathcal{B})$ we have $det\,(G(\theta)_{\bullet\mathcal{B}}) \neq 0$, under condition (1), an equivalent formulation for any invariancy region can be given by ensuring that the numerator of each defining rational inequality is nonnegative. Similarly, under condition (2), an equivalent formulation for any invariancy region can be given by ensuring that the numerator of each defining rational inequality is nonpositive. □

As a consequence of Lemma 5.15, for each complementary basis $\mathcal{B}$ we define the following:

$$g(\mathcal{B}) := \begin{cases} 1 & \text{if } det\,(G(\theta)_{\bullet\mathcal{B}}) \geq 0 \text{ for all } \theta \in S_\theta \\ -1 & \text{if } det\,(G(\theta)_{\bullet\mathcal{B}}) \leq 0 \text{ for all } \theta \in S_\theta \end{cases} \tag{5.9}$$

Then for each complementary basis $\mathcal{B}$ the associated invariancy region can be expressed with polynomial defining inequalities as

$$\mathcal{IR}_{\mathcal{B}} := \{\theta \in \mathcal{P}(\mathcal{B}) : g(\mathcal{B}) Adj\,(G(\theta)_{\bullet\mathcal{B}})\, q(\theta) \geq 0\}. \tag{5.10}$$

We now recall the following definition from [14].

Definition 5.17. A *semi-algebraic subset* of $\mathbb{R}^n$ is a subset of the form

$$\bigcup_{i=1}^{s} \bigcap_{j=1}^{r_i} \{x \in \mathbb{R}^n : f_{i,j}(x) \star_{i,j} 0\} \tag{5.11}$$

where, for each $i \in \{1, \dots, s\}$ and $j \in \{1, \dots, r_i\}$, $f_{i,j}$ is a polynomial function on $\mathbb{R}^n$ and $\star_{i,j}$ represents either "=" or "<."

Hence, from Proposition 5.16 we are able to make the following observation.

Observation 5.18. Every invariancy region is a semi-algebraic subset of S_θ.

Return to Examples 5.5 and 5.6 and observe the invariancy regions for $\mathcal{B}_0^{5.5}$ and $\mathcal{B}_0^{5.6}$ expressed as semi-algebraic sets:

$$\mathcal{IR}_{\mathcal{B}_0^{5.5}} = \left\{ \theta \in \mathcal{P}(\mathcal{B}_0^{5.5}) : \begin{array}{c} 3\theta_1^3 + 18\theta_1^2\theta_2 - 49\theta_1^2 - 75\theta_1\theta_2^2 + 148\theta_1\theta_2 + 68\theta_1 + 96\theta_2^2 - 16\theta_2 - 76 \leq 0 \\ -(\theta_1+2)(9\theta_1^3 - 9\theta_1^2\theta_2 - 33\theta_1^2 - 87\theta_1\theta_2^2 + 21\theta_1\theta_2 + 22\theta_1 - 59\theta_2^3 + 13\theta_2^2 + 50\theta_2 + 5) \leq 0 \\ (\theta_1+2)(-6\theta_1^2 - \theta_1\theta_2 + 11\theta_1 + 15\theta_2^2 - 16\theta_2 + 1) \leq 0 \\ (\theta_1+2)(3\theta_1^2 + 8\theta_1\theta_2 - \theta_1 + 5\theta_2^2 + 5\theta_2 - 11) \leq 0 \\ (\theta_1+2)(9\theta_1 - 13\theta_2 + \theta_1\theta_2 + 21\theta_2^2 - 12) \leq 0 \end{array} \right\} \tag{5.12}$$

$$\mathcal{IR}_{\mathcal{B}_0^{5.6}} = \left\{ \theta \in \mathcal{P}(\mathcal{B}_0^{5.6}) : \begin{array}{c} -\theta_2 - 1 \geq 0 \\ \theta_1 - \theta_2 - 1 \geq 0 \\ -18\theta_2 - 34 \geq 0 \\ -9\theta_2 - 17 \geq 0 \end{array} \right\} \tag{5.13}$$

5.2.3 Geometry of the mpLCP

In this subsection we discuss some of the properties of the (mp)LCP problem that we will need in order to establish an algorithm for partitioning S_θ. We now give several more definitions and properties needed for this discussion.

Definition 5.19. For an index $i \in \mathcal{E}$ the *complementary index* of i is $\bar{\imath} := (i+h) \mod 2h$.

Similarly, given $\mathcal{I} \subseteq \mathcal{E}$, the set of all complementary indices of elements in $\mathcal{I}$ is denoted $\overline{\mathcal{I}}$.

Definition 5.20. A set $\mathcal{J} \subset \mathcal{E}$ is called *complementary* if $|\{i, i+h\} \cap \mathcal{J}| = 1 \ \forall i \in \{1, \ldots, h\}$, i.e., if $i \in \mathcal{J} \Rightarrow \bar{\imath} \notin \mathcal{J}$.

Given an arbitrary matrix Q, we use the notation *cone*(Q) to represent the set of all nonnegative combinations of the columns of Q.

Definition 5.21. For any complementary set $\mathcal{J}$ and any $\theta \in S_\theta$, the set $\mathcal{C}(\theta, \mathcal{J}) := cone\left(G(\theta)_{\bullet\mathcal{J}}\right)$ is called a *parametric complementary cone* with respect to the matrix $M(\theta)$.

Definition 5.22. A parametric complementary cone $\mathcal{C}(\theta, \mathcal{J})$ is *full-dimensional* if $dim(\mathcal{C}(\theta, \mathcal{J})) = h$, i.e., if $rank\left(G(\theta)_{\bullet\mathcal{J}}\right) = h$.

Proposition 5.23. A parametric complementary cone $\mathcal{C}(\theta, \mathcal{J})$ is full-dimensional if and only if $\mathcal{J}$ is a complementary basis and $\theta \in \mathcal{P}(\mathcal{J})$.

Proof. ($\Rightarrow$): Since $\mathcal{C}(\theta, \mathcal{J})$ is a parametric complementary cone, $\mathcal{J}$ is complementary and thus $|\mathcal{J}| = h$. Since $\mathcal{C}(\theta, \mathcal{J})$ is full-dimensional, $rank\left(G(\theta)_{\bullet\mathcal{J}}\right) = h$ and thus $\theta \in \mathcal{P}(\mathcal{J})$.

($\Leftarrow$): Since $\mathcal{J}$ is a complementary basis and $\theta \in \mathcal{P}(\mathcal{J})$, we have $rank\left(G(\theta)_{\bullet\mathcal{J}}\right) = h$. Thus, $dim\left(\mathcal{C}(\theta, \mathcal{J})\right) = h$. □

Definition 5.24. For distinct complementary bases $\mathcal{B}_1$ and $\mathcal{B}_2$ and fixed $\theta \in S_\theta$, the parametric complementary cones $\mathcal{C}(\theta, \mathcal{B}_1)$ and $\mathcal{C}(\theta, \mathcal{B}_2)$ are called *adjacent* if $dim\left(\mathcal{C}(\theta, \mathcal{B}_1) \cap \mathcal{C}(\theta, \mathcal{B}_2)\right) = h-1$. In this case the bases $\mathcal{B}_1$ and $\mathcal{B}_2$ are also called adjacent.

For a complementary basis $\mathcal{B}$ and $\theta \in S_\theta$, the associated parametric complementary cone is:

$$\mathcal{C}(\theta, \mathcal{B}) = \left\{\tau \in \mathbb{R}^h : G(\theta)_{\bullet\mathcal{B}}^{-1}\tau \geq 0\right\} \tag{5.14}$$

Thus, given some $\theta \in S_\theta$, $\mathcal{C}\left(\theta, \mathcal{B}\right)$ defines the set of all vectors for which the basis $\mathcal{B}$ is feasible. Similarly, $\mathcal{K}(M(\theta)) := \cup_{\text{feasible bases } \mathcal{B}} \mathcal{C}\left(\theta, \mathcal{B}\right)$ defines the set of all vectors for which any basis is feasible. We now introduce another definition.

Definition 5.25. Given a feasible complementary basis $\mathcal{B}$, the *$\mathcal{B}$-complementary subspace* of $\mathbb{R}^h$ is the set

$$\mathcal{CS}(\mathcal{B}) = \bigcup_{\theta \in S_\theta} \mathcal{C}(\theta, \mathcal{B}) = \left\{\tau \in \mathbb{R}^h : \exists\, \theta \in S_\theta \text{ s.t. } G(\theta)^{-1}_{\bullet\mathcal{B}} \tau \geq 0\right\}. \tag{5.15}$$

Note the close relationship between $\mathcal{IR}_\mathcal{B}$ and $\mathcal{CS}\left(\mathcal{B}\right)$. In order to understand this relationship further, we now consider the specifics of the following set:

$$\mathcal{AS} := \left\{\tau \in \mathbb{R}^h : \exists\, \theta \in S_\theta \text{ s.t. } \tau = q(\theta)\right\}. \tag{5.16}$$

Since every element of the vector $q(\theta)$ is an affine function of θ, there exists a vector $q \in \mathbb{R}^h$ and a matrix $\triangle Q \in \mathbb{R}^{h \times k}$ such that $q(\theta) = q + \triangle Q \theta$. When considering $\mathcal{AS}$ we make the assumption that $\triangle Q$ is of full column rank. In the case when $\triangle Q$ is not full column rank, there exists a "reduced" mpLCP which can be solved in place of the original mpLCP. Details on this reduced mpLCP can be found in Section 4.2.2.2.

We continue our study of $\mathcal{AS}$ by introducing another definition.

Definition 5.26. Given a complementary basis $\mathcal{B}$, the *invariant domain of* $\mathcal{B}$, denoted $\mathcal{ID}_\mathcal{B}$, is the intersection of the subspace $\mathcal{AS}$ with the $\mathcal{B}$-complementary subspace $\mathcal{CS}(\mathcal{B})$. Thus $\mathcal{ID}_\mathcal{B} := \mathcal{CS}(\mathcal{B}) \cap \mathcal{AS} = \left\{\tau : G^{-1}_{\bullet\mathcal{B}} \tau \geq 0, \tau = q(\theta), \theta \in S_\theta\right\}$.

Since we have assumed that $\triangle Q$ is of full column rank, we can view $q(\theta)$ as a bijective function with inverse $q^{-1}(\cdot)$. As a result, we make the following two observations:

Observation 5.27. For a given basis $\mathcal{B}$, we have $\mathcal{ID}_\mathcal{B} = q\left(\mathcal{IR}_\mathcal{B}\right)$ and $\mathcal{IR}_\mathcal{B} = q^{-1}(\mathcal{ID}_\mathcal{B})$.

Observation 5.28. For each $i \in \mathcal{E}$, the inequality $\left(G(\theta)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} \tau \geq 0$ is redundant in $\mathcal{ID}_\mathcal{B}$ if and only if $\left(G(\theta)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\theta) \geq 0$ is redundant in $\mathcal{IR}_\mathcal{B}$.

Notice that, given a basis $\mathcal{B}$, these observations show explicitly the relationship existing between the associated invariancy region and the $\mathcal{B}$-complementary subspace.

Observe the following definition which we employ when dealing with convex sets.

Definition 5.29. For an arbitrary convex set S, the relative interior of S is the set $relint(S) := \{s \in S : \exists \epsilon > 0,\, B_\epsilon(s) \cap aff(S) \subseteq S\}$, where $B_\epsilon(s)$ is the ball of radius ϵ centered at s and $aff(S)$ is the affine hull of S, i.e., the intersection of all affine sets containing S.

Briefly consider non-parametric LCP. In this context we use the notations M and $\mathcal{K}(M)$ to denote the non-parametric counterparts of $M(\theta)$ and $\mathcal{K}(M(\theta))$ and the term *complementary cone* to refer to the non-parametric counterpart of a parametric complementary cone. It is well known that given an instance of LCP, and by extension mpLCP, the properties of the matrix M have a significant impact on the structure of the complementary cones, and hence also the parametric complementary cones and invariancy regions in the case of mpLCP. Of particular importance in this work are column sufficient matrices, which we defined in Definition 5.1, and Q_0 matrices, for which we quote the definition from [65].

Definition 5.30. The square matrix M is said to be a Q_0 matrix if $\mathcal{K}(M)$ is a convex cone.

We now recall the following important property of column sufficient matrices, shown in [22]: if M is column sufficient, all complementary cones have disjoint relative interiors, i.e., the complementary cones partition $\mathcal{K}(M)$. The theory we develop throughout the rest of this work and the algorithms we present in Sections 5.4 and 5.5 rely heavily on these properties of column sufficient and Q_0 matrices. As a result, the algorithms we propose are designed for instances of mpLCP in which $M(\theta)$ is both Q_0 and column sufficient for all $\theta \in S_\theta$. Note that the largest class of matrices known to be a subset of both the classes of Q_0 and column sufficient matrices is the class of sufficient matrices.

We now further consider invariancy regions and establish properties of these regions which allow us to show that a partition of S_θ exists and has a particular structure.

5.3 Algebraic Properties of Invariancy Regions

In this section we merge several concepts from the fields of Algebraic Geometry, Algebraic Topology and Operations Research. We point out that although the details presented here are theoretically important and provide the necessary foundation for the development of the algorithms presented in Sections 5.4 and 5.5, they are not necessary for the understanding of later sections and may be skipped by the uninterested reader.

5.3.1 Decomposition of the Parameter Space

We continue our discussion of invariancy regions by examining the properties of various subsets of invariancy regions as well as S_θ and the subspace $\mathcal{AS}$. Consider an arbitrary element $M(\theta)_{ij}$ of the matrix $M(\theta)$. Recognize that since this element is an affine function of θ, it can be represented as $M(\theta)_{ij} = \sum\limits_{n=1}^{k} \alpha_{ijn}\theta_n + \beta_{ij}$. Define the following index sets: (i) $U = \{n \in \{1, \dots, k\} : \alpha_{ijn} \neq 0 \text{ for some } i, j \in \{1, \dots, h\}\}$, and (ii) $V = \{i, \dots, k\} \setminus U$. Using these index sets we define $\phi := \theta_U$ and $\sigma := \theta_V$. Then ϕ is the subvector of θ such that every element of ϕ is present in some element of $M(\theta)$ and σ is the subvector of θ such that no element of σ is present in any element of $M(\theta)$. We let p represent the dimension of ϕ, and consequently the dimension of σ is $k - p$. Note that we allow for the case in which $p = k$. We do assume, however, that $p \neq 0$, since this is precisely the case dealt with in Adelgren and Wiecek [2]. We now define

$$S_\phi(\sigma) = \{\phi^* \in \mathbb{R}^p : \theta^* = (\phi^*, \sigma) \in S_\theta\} \tag{5.17}$$

and

$$S_\sigma(\phi) = \{\sigma^* \in \mathbb{R}^{k-p} : \theta^* = (\phi, \sigma^*) \in S_\theta\}. \tag{5.18}$$

$S_\phi(\sigma)$ can be interpreted as the set of attainable values of ϕ given a fixed σ and similarly, $S_\sigma(\phi)$ can be interpreted as the set of attainable values of σ given a fixed ϕ. From these sets we can also define

$$\begin{aligned} S_\phi &= \{\phi \in \mathbb{R}^p : S_\sigma(\phi) \neq \emptyset \text{ for some } \sigma \in \mathbb{R}^{k-p}\} \\ &= \operatorname{Proj}_\phi S_\theta \end{aligned} \tag{5.19}$$

and

$$\begin{aligned} S_\sigma &= \{\sigma \in \mathbb{R}^{k-p} : S_\phi(\sigma) \neq \emptyset \text{ for some } \phi \in \mathbb{R}^p\} \\ &= \operatorname{Proj}_\sigma S_\theta. \end{aligned} \tag{5.20}$$

Now, given a basis $\mathcal{B}$ and some $\theta = (\phi, \sigma) \in \mathcal{P}(\mathcal{B})$, we can use these sets to define the following subsets of the invariancy region $\mathcal{IR}_\mathcal{B}$:

$$\mathcal{IR}^\phi_\mathcal{B}(\sigma) = \{\phi^* \in \mathbb{R}^p : \theta^* = (\phi^*, \sigma) \in \mathcal{IR}_\mathcal{B}\} \tag{5.21}$$

$$\mathcal{IR}^{\sigma}_{\mathscr{B}}(\phi) = \{\sigma^* \in \mathbb{R}^{k-p} : \theta^* = (\phi, \sigma^*) \in \mathcal{IR}_{\mathscr{B}}\} \tag{5.22}$$

$$\begin{aligned} \mathcal{IR}^{\phi}_{\mathscr{B}} &= \{\phi \in \mathbb{R}^{p} : \mathcal{IR}^{\sigma}_{\mathscr{B}}(\phi) \neq \emptyset \text{ for some } \sigma \in \mathbb{R}^{k-p}\} \\ &= \text{Proj}_{\phi}\, \mathcal{IR}_{\mathscr{B}} \end{aligned} \tag{5.23}$$

$$\begin{aligned} \mathcal{IR}^{\sigma}_{\mathscr{B}} &= \{\sigma \in \mathbb{R}^{k-p} : \mathcal{IR}^{\phi}_{\mathscr{B}}(\sigma) \neq \emptyset \text{ for some } \phi \in \mathbb{R}^{p}\} \\ &= \text{Proj}_{\sigma}\, \mathcal{IR}_{\mathscr{B}} \end{aligned} \tag{5.24}$$

Recognize that $\mathcal{IR}^{\phi}_{\mathscr{B}}(\sigma)$ and $\mathcal{IR}^{\sigma}_{\mathscr{B}}(\phi)$ can be interpreted as "cross-sections" of $\mathcal{IR}_{\mathscr{B}}$ associated with fixing a particular $\sigma \in S_\sigma$ or $\phi \in S_\phi$, respectively. Additionally, $\mathcal{IR}^{\phi}_{\mathscr{B}}$ and $\mathcal{IR}^{\sigma}_{\mathscr{B}}$ represent the projections of $\mathcal{IR}_{\mathscr{B}}$ onto S_ϕ and S_σ, respectively. As a result, we make the following observations.

Observation 5.31. The sets $\mathcal{IR}^{\phi}_{\mathscr{B}}$ and $\mathcal{IR}^{\sigma}_{\mathscr{B}}$ are semi-algebraic.

Observation 5.32. For each $\sigma \in \mathcal{IR}^{\sigma}_{\mathscr{B}}$, the set $\mathcal{IR}^{\phi}_{\mathscr{B}}(\sigma)$ is semi-algebraic.

Observation 5.33. For each $\phi \in \mathcal{IR}^{\phi}_{\mathscr{B}}$, the set $\mathcal{IR}^{\sigma}_{\mathscr{B}}(\phi)$ is polyhedral.

Having now defined the subvectors ϕ and σ, recognize that since $M(\theta)$ contains no elements of σ, we can write $M(\theta)$ as $M(\phi)$ and $G(\theta)$ as $G(\phi)$. Similarly, for any basis $\mathscr{B}$ we can write $\mathcal{C}(\theta, \mathscr{B})$ as $\mathcal{C}(\phi, \mathscr{B})$. Next consider the vector $q(\theta)$. Recall from our discussion in Section 5.2.3 that it can be represented as $q(\theta) = q + \triangle Q\theta$. Using the notation introduced above, we can also represent $q(\theta)$ as $q(\phi, \sigma) = q + \triangle Q_{\bullet U}\phi + \triangle Q_{\bullet V}\sigma$. Finally, for a fixed $\phi \in S_\phi$ we introduce the following subset of the subspace $\mathscr{AS}$:

$$\mathscr{AS}(\phi) := \{\tau \in \mathbb{R}^h : \tau = q(\phi, \sigma), \sigma \in S_\sigma\}. \tag{5.25}$$

Consider the following results.

Definition 5.34. (Definition 2.2.5 of [14]) Let $A \in \mathbb{R}^n$ and $B \in \mathbb{R}^m$ be two semi-algebraic sets. A mapping $f : A \to B$ is *semi-algebraic* if its graph is semi-algebraic in $\mathbb{R}^{m+n}$.

Lemma 5.35. (Theorem 2.8.8 of [14]) Let A be a semi-algebraic set and $f : A \to \mathbb{R}^n$ a semi-algebraic mapping. Then $dim(A) \geq dim(f(A))$. If f is a bijection from A onto $f(A)$, then $dim(A) = dim(f(A))$.

Proposition 5.36. The following hold for each $\phi \in S_\phi$:

1. $dim(\mathcal{AS}(\phi)) = dim(S_\sigma)$

2. $dim(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})) = dim(\mathcal{IR}^\sigma_\mathcal{B}(\phi))$

Proof. Recall that in this work we make the assumption that $\triangle Q$ is of full column rank. Hence, $\triangle Q_{\bullet V}$ must also be of full column rank. As a result, for each fixed $\phi \in S_\phi$, $q(\phi, \sigma)$ can be viewed as a bijective function of σ. Furthermore, for each fixed $\phi \in S_\phi$, $q(\phi, \sigma)$ is also semi-algebraic since its graph is the set $\{(\sigma, q') : \sigma \in S_\sigma, q' = q + \triangle Q_{\bullet U}\phi + \triangle Q_{\bullet V}\sigma\}$, which is clearly semi-algebraic since S_σ is polyhedral. Thus, by Lemma 5.35, the above hold. □

Using the sets and notations introduced so far in this section, together with (5.9), for each feasible complementary basis $\mathcal{B}$ we provide the following alternate forms of the definitions of invariancy regions and invariant domains:

$$\mathcal{IR}_\mathcal{B} := \{(\phi, \sigma) \in \mathcal{P}(\mathcal{B}) : g(\mathcal{B}) Adj\left(G(\phi)_{\bullet\mathcal{B}}\right) q(\phi, \sigma) \geq 0\} \tag{5.26}$$

$$\mathcal{ID}_\mathcal{B} = \bigcup_{\phi \in S_\phi} \left(\mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{AS}(\phi)\right) \tag{5.27}$$

Recognize that the definition of an invariancy region in (5.26), together with the results of Propositions 5.14, **??**, 5.16 and Lemma 5.15 and the facts that the elements of $M(\phi)$ are affine functions of ϕ and the elements of $q(\phi, \sigma)$ are affine functions of ϕ and σ, leads to the following observation.

Observation 5.37. For each feasible complementary basis $\mathcal{B}$, the defining inequalities of the invariancy region $\mathcal{IR}_\mathcal{B}$ are polynomial in ϕ and linear in σ.

Throughout the remainder of this section we develop theoretical results that allow us to exploit the property of invariancy regions described in Observation 5.37. Although this property was discovered in the context of invariancy regions, it is of interest to study in general. For this reason we establish the following property.

Property 5.38. Given a subset S' of S_θ, we say S' satisfies Property 5.38 if the following hold:

- S' is defined by a set of inequalities $f_i(\phi, \sigma) \leq 0$, $i \in \{1, \ldots, n\}$ for some $n \in \mathbb{N}$.

- For each $i \in \{1, \ldots, n\}$, f_i is polynomial in ϕ and linear in σ.

In the following subsection we consider general sets satisfying Property 5.38, develop interesting theoretical consequences of this properties, and use these results to make important conclusions about invariancy regions.

5.3.2 Exploiting the Algebraic Structure of an Invariancy Region

Perhaps the most important aspect of invariancy regions for us to study is their dimension. In order to establish the existence of a partition of S_θ we must develop necessary and sufficient conditions for an invariancy region to be of dimension k or $k-1$. To do this, however, we must first establish many more properties of invariancy regions or, more generally, subsets of S_θ having Property 5.38. In the following discussion we use Ω to denote an arbitrary subset of S_θ having Property 5.38. We introduce several new definitions and notations as well as many new propositions which we develop by exploiting this property.

Definition 5.39. (Kalajdzievski [49]) Two subsets A and B of a space Y are *ambient isotopic within* Y if there is a continuous mapping $H : Y \times [0,1] \to Y$ such that the mappings $H_t : Y \to Y, t \in [0,1]$, satisfy the following two conditions:

(i) H_t is a homeomorphism for every $t \in [0,1]$.

(ii) $H_0 = id_Y$ (the identity map) and $H_1(A) = B$.

Definition 5.40. (Definition 2.4.2 of Bochnak et al. [14]) A semi-algebraic subset A of $\mathbb{R}^n$ is *semi-algebraically connected* if for every pair of semi-algebraic sets F_1 and F_2 in A, disjoint and satisfying $F_1 \cup F_2 = A$, one has $F_1 = A$ or $F_2 = A$.

Definition 5.41. (Definition 2.5.12 of Bochnak et al. [14]) A semi-algebraic subset A of $\mathbb{R}^n$ is *semi-algebraically path connected* if, for every $x, y \in A$, there exists a continuous semi-algebraic mapping $\psi : [0,1] \to A$ such that $\psi(0) = x$ and $\psi(1) = y$.

Proposition 5.42. (Proposition 2.8.5 of Bochnak et al. [14])

1. Let $A = \bigcup_{i=1}^n A_i$ be a finite union of semi-algebraic sets. Then $dim(A) = \max(dim(A_1), \dots, dim(A_n))$.

2. Let A and B be two semi-algebraic sets. Then $dim(A \times B) = dim(A) + dim(B)$.

We now introduce new notation that we will use to show that for any $\Omega \subset S_\theta$ having Property 5.38, the dimension of Ω can be expressed in terms of the dimensions of certain subsets of Ω which arise due to the decomposition of θ into the subvectors ϕ and σ. In a similar fashion to the definitions given in (5.21), (5.22), (5.23) and (5.24) we introduce the following sets.

$$\Omega^{\phi}(\sigma) = \{\phi \in \mathbb{R}^p : \theta = (\phi, \sigma) \in \Omega\} \tag{5.28}$$

$$\Omega^{\sigma}(\phi) = \{\sigma \in \mathbb{R}^{k-p} : \theta = (\phi, \sigma) \in \Omega\} \tag{5.29}$$

$$\begin{aligned} \Omega^{\phi} &= \{\phi \in \mathbb{R}^p : \Omega^{\sigma}(\phi) \neq \emptyset \text{ for some } \sigma \in \mathbb{R}^{k-p}\} \\ &= \operatorname{Proj}_{\phi} \Omega \end{aligned} \tag{5.30}$$

$$\begin{aligned} \Omega^{\sigma} &= \{\sigma \in \mathbb{R}^{k-p} : \Omega^{\phi}(\sigma) \neq \emptyset \text{ for some } \phi \in \mathbb{R}^p\} \\ &= \operatorname{Proj}_{\sigma} \Omega \end{aligned} \tag{5.31}$$

We now define the following sets which play an important role throughout this section.

$$\Phi_{\Omega}^{(d)} := \{\phi \in \Omega^{\phi} : dim(\Omega^{\sigma}(\phi)) = d\} \tag{5.32}$$

$$\mathcal{D}_{\Omega} := \text{The set of defining inequalities of } \Omega. \tag{5.33}$$

$$\mathscr{D}_{\Omega} := \text{The set of subsets of } \mathcal{D}_{\Omega}. \tag{5.34}$$

Then for each $\mathcal{S} \in \mathscr{D}_{\Omega}$, let

$$\Phi_{\Omega,\mathcal{S}} := \{\phi \in \Omega^{\phi} : \text{The set of redundant inequalities of } \Omega^{\sigma}(\phi) \text{ is } \mathcal{D}_{\Omega} \setminus \mathcal{S}\}. \tag{5.35}$$

For each $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_{\Omega}$, let

$$\Phi_{\Omega,\mathcal{S}}^{(d)} := \Phi_{\Omega}^{(d)} \cap \Phi_{\Omega,\mathcal{S}} \tag{5.36}$$

and

$$\mathscr{T}_{\Omega,\mathcal{S}}^{(d)} := \{(\phi, \sigma) \in \Omega : \phi \in \Phi_{\Omega,\mathcal{S}}^{(d)}, \sigma \in \Omega^{\sigma}(\phi)\}. \tag{5.37}$$

From (5.8), (5.22) and (5.37) we make the following observation.

Observation 5.43. For any set $\Omega \subset S_\theta$ having Property 5.38, we have $\Omega = \bigcup_{d \in \mathbb{N}, \mathcal{S} \in \mathscr{D}_\Omega} \mathscr{T}^{(d)}_{\Omega,\mathcal{S}}$.

We now introduce several theoretical results which provide us with a strategy for determining the dimension of an invariancy region and moreover, establishing necessary and sufficient conditions for an invariancy region to have dimension k or $k-1$.

Proposition 5.44. Given $\Omega \subset S_\theta$ having Property 5.38, the set $\Phi^{(d)}_\Omega$ is semi-algebraic for any $d \in \mathbb{N}$.

Proof. We proceed by showing that the set $\Phi^{(\geq d)}_\Omega := \{\phi \in \Omega^\phi : dim(\Omega^\sigma(\phi)) \geq d\}$ is semi-algebraic for each $d \in \mathbb{N}$. This clearly implies that $\Phi^{(d)}_\Omega$ is semi-algebraic since $\Phi^{(d)}_\Omega = \Phi^{(\geq d)}_\Omega \setminus \Phi^{(\geq d+1)}_\Omega$ and intersections and complements of semi-algebraic sets are also semi-algebraic.

We now construct a set $\mathcal{X}^d_\Omega$ such that $\Phi^{(\geq d)}_\Omega$ is the projection of $\mathcal{X}^d_\Omega$ onto S_ϕ. We then show that $\mathcal{X}^d_\Omega$ is semi-algebraic. This is enough to show that $\Phi^{(\geq d)}_\Omega$ is semi-algebraic since projections of semi-algebraic sets are also semi-algebraic. We define $\mathcal{X}^d_\Omega$ to be the set of $(d+2)$-tuples $(\phi, \sigma_1, \ldots, \sigma_{d+1})$ which satisfy the following conditions:

1. For each $i \in \{1, \ldots, d+1\}$, $(\phi, \sigma_i) \in \Omega$.
2. The matrix $K^{\sigma_1,\ldots,\sigma_{d+1}}$ is of full rank, where $K^{\sigma_1,\ldots,\sigma_{d+1}} \in \mathbb{R}^{(k-p)\times d}$ is defined so that for each $i \in \{1, \ldots, d\}$, $K^{\sigma_1,\ldots,\sigma_{d+1}}_{\bullet i} = \sigma_i - \sigma_{d+1}$.

Recognize that condition (1) is enforced by a set of polynomial constraints since Ω is semi-algebraic. Also recognize that condition (2) is satisfied if and only if there exists a $d \times d$ minor of $K^{\sigma_1,\ldots,\sigma_{d+1}}$ whose determinant is nonzero. Denote the $d \times d$ minors of $K^{\sigma_1,\ldots,\sigma_{d+1}}$ as $D_1, \ldots, D_n$ (clearly n is finite). Then condition (2) is satisfied if and only if at least one of $D_1, \ldots, D_n$ has a nonzero determinant. Recognize, though, that at least one of $D_1, \ldots, D_n$ has a nonzero determinant if and only if there exists a $y \in \mathbb{R}$ such that the following system has a solution:

$$\begin{aligned} y &> 0 \\ y\,(det\,(D_i))^2 &< 1 \text{ for each } i \in \{1, \ldots, n\} \\ \prod_{i=1}^{n} \left(1 - y\,(det\,(D_i))^2\right) &< 1. \end{aligned}$$

Since the determinant of a matrix can be expressed as a polynomial function of its elements, this shows that condition (2) is also enforced by a set of polynomial constraints. Hence, $\mathcal{X}^d_\Omega$ is semi-algebraic. Further notice that conditions (1) and (2) ensure that a $(d+2)$-tuple $(\phi, \sigma_1, \ldots, \sigma_{d+1})$ is in $\mathcal{X}_\Omega^d$ if and only if: (i) $\phi \in \Omega^\phi$, (ii) each $\sigma_i \in \Omega^\sigma(\phi)$, and (iii) $\sigma_1, \dots, \sigma_{d+1}$ are affinely independent. Thus, the projection of $\mathcal{X}_\Omega^d$ onto S_ϕ is the set of all ϕ in Ω^ϕ such that the dimension of $\Omega^\sigma(\phi)$ is at least d, which is precisely $\Phi_\Omega^{(\geq d)}$. □

Proposition 5.45. Given $\Omega \subset S_\theta$ having Property 5.38, the set $\Phi_{\Omega,\mathcal{S}}$ is semi-algebraic for any $\mathcal{S} \in \mathscr{D}_\Omega$.

Proof. Consider $\overline{\Phi_{\Omega,\mathcal{S}}} := \{\phi \in \Omega^\phi : \text{The inequalities of } \mathcal{S} \text{ are not redundant in } \Omega^\sigma(\phi)\}$. We proceed by showing that $\overline{\Phi_{\Omega,\mathcal{S}}}$ is semi-algebraic for each $\mathcal{S} \in \mathscr{D}_\Omega$. This clearly implies that $\Phi_{\Omega,\mathcal{S}}$ is semi-algebraic since $\Phi_{\Omega,\mathcal{S}} = \overline{\Phi_{\Omega,\mathcal{S}}} \setminus \left(\cup_{\mathcal{E}\in\mathscr{D}_\Omega:\mathcal{E}\supset\mathcal{S}}\overline{\Phi_{\Omega,\mathcal{E}}}\right)$ and unions, intersections and complements of semi-algebraic sets are also semi-algebraic.

We now construct a set $\mathcal{Z}_{\Omega,\mathcal{S}}$ such that $\Phi_{\Omega,\mathcal{S}}$ is the projection of $\mathcal{Z}_{\Omega,\mathcal{S}}$ onto S_ϕ. We then show that $\mathcal{Z}_{\Omega,\mathcal{S}}$ is semi-algebraic. This is enough to show that $\Phi_{\Omega,\mathcal{S}}$ is semi-algebraic since projections of semi-algebraic sets are also semi-algebraic. Given $\mathcal{S} \in \mathscr{D}_\Omega$, we define $\mathcal{Z}_{\Omega,\mathcal{S}}$ to be the set of $(|\mathcal{S}|+1)$-tuples $(\phi, \sigma_1, \dots, \sigma_{|\mathcal{S}|})$ which satisfy the following conditions:

1. For each $i \in \{1, \dots, |\mathcal{S}|\}$, $(\phi, \sigma_i) \in \Omega$.

2. For each inequality $\mathcal{S}_i \in \mathcal{S}$, (ϕ, σ_i) satisfies $\mathcal{S}_i$ at equality and satisfies each inequality in $\mathcal{D}_\Omega \setminus \mathcal{S}_i$ strictly.

As it is clear that conditions (1) and (2) are each enforced by sets of polynomial inequalities, $\mathcal{Z}_{\Omega,\mathcal{S}}$ is semi-algebraic. Furthermore, recognize that condition (2) is satisfied if and only if $\mathcal{S}$ contains no constraints which are redundant in Ω. Hence, the projection of $\mathcal{Z}_{\Omega,\mathcal{S}}$ onto S_ϕ is exactly $\overline{\Phi_{\Omega,\mathcal{S}}}$. □

Corollary 5.46. If $\Omega \subset S_\theta$ satisfies Property 5.38, the set $\Phi_{\Omega,\mathcal{S}}^{(d)}$ is semi-algebraic for any $\mathcal{S} \in \mathscr{D}_\Omega$ and $d \in \mathbb{N}$.

Proof. The result follows directly from the proofs of Propositions 5.44 and 5.45 and the fact that intersections of semi-algebraic sets are also semi-algebraic. □

Proposition 5.47. If $\Omega \subset S_\theta$ satisfies Property 5.38, for any $\phi^* \in \Phi_{\Omega,\mathcal{S}}^{(d)}$, the set $\mathscr{T}_{\Omega,\mathcal{S}}^{(d)}$ is ambient isotopic to

$$\mathscr{U}_{\Omega,\mathcal{S}}^{(d)}(\phi^*) := \{(\phi, \sigma) \in \Omega : \phi \in \Phi_{\Omega,\mathcal{S}}^{(d)}, \sigma \in \Omega^\sigma(\phi^*)\}. \tag{5.38}$$

Proof. Recognize from (5.36), (5.35) and (5.32) that $\Phi_{\Omega,\mathcal{S}}^{(d)}$ is the set of $\phi \in \Omega^\phi$ such that $\Omega^\sigma(\phi)$ is invariant with respect to dimension and the set of defining inequalities. By Corollary 5.46, the set

$\Phi^{(d)}_{\Omega,\mathcal{S}}$ is semi-algebraic and is therefore composed of a finite number of semi-algebraically connected subsets. For the remainder of this discussion we assume without loss of generality that $\Phi^{(d)}_{\Omega,\mathcal{S}}$ is a single semi-algebraically connected set. If not, the arguments that follow can be applied iteratively to each of its semi-algebraically connected subsets.

We proceed with the proof by: (i) establishing the concepts which are used to define a mapping $\mathscr{H}(\cdot)$ which we use to show that $\mathscr{T}^{(d)}_{\Omega,\mathcal{S}}$ is ambient isotopic to $\mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi^*)$, (ii) explicitly defining the mapping $\mathscr{H}(\cdot)$, and (iii) arguing that $\mathscr{H}(\cdot)$ satisfies the properties outlined in Definition 5.39.

Since Ω satisfies Property 5.38, recognize that $\Omega^\sigma(\phi)$ is a d-dimensional polyhedron for every $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. Therefore $\Omega^\sigma(\phi)$ must have a finite number of extreme points for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. Furthermore, since the set of non-redundant inequalities of $\Omega^\sigma(\phi)$ is invariant for all $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, given a $\phi' \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, a subset $\mathcal{S}'$ of d inequalities in $\mathcal{S}$ intersect at an extreme point of $\mathcal{IR}^\sigma_\Omega(\phi')$ if and only if the inequalities in $\mathcal{S}'$ intersect at an extreme point of $\mathcal{IR}^\sigma_\Omega(\phi)$ for all $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. In this case we refer to the set $\mathcal{S}' \subset \mathcal{S}$ as an *extreme point defining set* for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. For each $\mathcal{S} \in \mathscr{D}_\Omega$ define the set

$$\mathcal{E}_{\mathcal{S}} := \{\mathcal{S}' \subset \mathcal{S} : |\mathcal{S}'| = d, \mathcal{S}' \text{ is an extreme point defining set for each } \phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}\}. \tag{5.39}$$

Recognize that for each $\mathcal{S} \in \mathscr{D}_\Omega$, $|\mathcal{E}_{\mathcal{S}}|$ is finite. Hence, assume $|\mathcal{E}_{\mathcal{S}}| = \ell$ and enumerate the elements of $\mathcal{E}_{\mathcal{S}}$ as $\mathcal{S}'_1, \dots, \mathcal{S}'_\ell$. Further recognize that the number of extreme points of $\Omega^\sigma(\phi)$ is ℓ for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. Then for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and $j \in \{1, \dots, \ell\}$ denote the extreme point of $\Omega^\sigma(\phi)$ defined by $\mathcal{S}'_j \in \mathcal{E}_{\mathcal{S}}$ as σ^j_ϕ. We now show that for all $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, each $\sigma \in \Omega^\sigma(\phi)$ can be uniquely represented in terms of the extreme points of $\Omega^\sigma(\phi)$. Notice that since $dim(\Omega^\sigma(\phi)) = d$ and $\Omega^\sigma(\phi) = conv(\{\sigma^1_\phi, \dots, \sigma^\ell_\phi\})$, where $conv(\cdot)$ represents the convex hull, we have $\ell \geq d+1$. If $\ell = d+1$ then $conv(\{\sigma^1_\phi, \dots, \sigma^\ell_\phi\})$ is a simplex and each $\sigma \in \Omega^\sigma(\phi)$ has a unique representation as a convex combination of $\sigma^1_\phi, \dots, \sigma^\ell_\phi$. On the other hand, if $\ell > d+1$, there is no such unique representation, in general, but we can construct a unique representation using a properly defined subset of $\sigma^1_\phi, \dots, \sigma^\ell_\phi$. Recognize from Lemma 2.3 of [7] that for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, $\Omega^\sigma(\phi)$ can be partitioned into a finite set of simplices $\Delta_1(\phi), \dots, \Delta_n(\phi)$ such that:

1. For each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and each $i \in \{1, \dots, n\}$, the vertices of $\Delta_i(\phi)$ are $d+1$ affinely independent points from $\{\sigma^1_\phi, \dots, \sigma^\ell_\phi\}$.

2. For each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and distinct $i, j \in \{1, \dots, n\}$, if the intersection $\Delta_i(\phi) \cap \Delta_j(\phi)$ is not empty, then it is a proper common face of both $\Delta_i(\phi)$ and $\Delta_j(\phi)$.

Given $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, $\sigma \in \Omega^{\sigma}(\phi)$ and $i \in \{1,\dots,n\}$ define the set of coefficients which can be used to represent σ as a convex combination of the elements of $\Delta_i(\phi)$:

$$\Upsilon(\phi,\sigma,i) := \left\{\mu \in \mathbb{R}^{\ell} : \sigma = \sum_{j=1}^{\ell} \mu_j \sigma^j_{\phi}, \sigma \in \Delta_i(\phi)\right\}. \tag{5.40}$$

Recognize that for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and $\sigma \in \Omega^{\sigma}(\phi)$ there exists at least one $i \in \{1,\dots,n\}$ such that $\Upsilon(\phi,\sigma,i) \neq \emptyset$. Thus, for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and $\sigma \in \Omega^{\sigma}(\phi)$ we define

$$\Upsilon(\phi,\sigma) := \left\{\mu \in \mathbb{R}^{\ell} : \mu \in \Upsilon(\phi,\sigma,i), i = \min\{j : \Upsilon(\phi,\sigma,j) \neq \emptyset\}\right\}. \tag{5.41}$$

Further recognize that for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, $\sigma \in \Omega^{\sigma}(\phi)$, and $i \in \{1,\dots,n\}$ either $\Upsilon(\phi,\sigma,i) = \emptyset$ or $|\Upsilon(\phi,\sigma,i)| = 1$. Hence, for all $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and $\sigma \in \Omega^{\sigma}(\phi)$, $|\Upsilon(\phi,\sigma)| = 1$. For each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ and $\sigma \in \Omega^{\sigma}(\phi)$ we denote the single element of $\Upsilon(\phi,\sigma)$ as $\mu^{\phi,\sigma}$. Note that we use this designation even when $\ell = d+1$ because in this case the arguments and definitions above still hold with $n = 1$.

Before we introduce the mapping $\mathscr{H}$, note that since $\Phi^{(d)}_{\Omega,\mathcal{S}}$ is semi-algebraically connected, by Proposition 2.5.13 of [14], it is also semi-algebraically path connected. For each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ let $\psi_{\phi}(\cdot) : [0,1] \to \Phi^{(d)}_{\Omega,\mathcal{S}}$ denote a continuous semi-algebraic mapping with $\psi_{\phi}(0) = \phi$ and $\psi_{\phi}(1) = \phi^*$ as outlined in Definition 5.41.

We now define the mapping $\mathscr{H}$ which we use to show that $\mathscr{T}^{(d)}_{\Omega,\mathcal{S}}$ and $\mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi^*)$ are ambient isotopic. Let $\Lambda := \Phi^{(d)}_{\Omega,\mathcal{S}} \times \left(\cup_{\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}} \Omega^{\sigma}(\phi)\right)$ and consider

$$\begin{aligned}\mathscr{H}(\cdot,\cdot) : \Lambda \times [0,1] \to \Lambda \text{ such that } \mathscr{H}(\theta,t) &= \mathscr{H}(\phi,\sigma,t) \\ &= \mathscr{H}\left(\phi, \sum_{j=1}^{\ell} \mu^{\phi,\sigma}_j \sigma^j_{\phi}, t\right) \\ &= \left(\phi, \sum_{j=1}^{\ell} \mu^{\phi,\sigma}_j \sigma^j_{\psi_{\phi}(t)}\right)\end{aligned}$$

We show that $\mathscr{H}$ satisfies the conditions outlined in Definition 5.39. We explore these conditions one at a time, in the following order:

(i) $\mathscr{H}$ is continuous.

(ii) For each fixed $t \in [0,1]$, $\mathscr{H}(\cdot,t)$ is a homeomorphism, i.e. the following hold:

- $\mathscr{H}(\cdot, t)$ is continuous.
- $\mathscr{H}(\cdot, t)$ is a bijection.
- $\mathscr{H}(\cdot, t)$ has a continuous inverse.

(iii) $\mathscr{H}(\mathscr{T}^{(d)}_{\Omega,\mathcal{S}}, 0) = \mathscr{T}^{(d)}_{\Omega,\mathcal{S}}$.

(iv) $\mathscr{H}(\mathscr{T}^{(d)}_{\Omega,\mathcal{S}}, 1) = \mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi^*)$.

<u>$\mathscr{H}$ is continuous</u>: Throughout this discussion we utilize the so-called infinity norm, also known as the uniform norm. For every $p \in \mathbb{N}$, given a vector $v \in \mathbb{R}^p$, we denote the infinity norm of v as $\|v\|$. Recall the following useful property of the infinity norm:

$$\text{Given } v \in \mathbb{R}^p \text{ and } \epsilon > 0, \|v\| \leq \epsilon \Leftrightarrow \|u\| \leq \epsilon \text{ for all subvectors } u \text{ of } v. \tag{5.42}$$

We will show that $\mathscr{H}$ is continuous by showing that for any $\epsilon > 0$ there exists $\delta(\epsilon) > 0$ such that, if $\left\|(\theta^1, t_1) - (\theta^2, t_2)\right\| < \delta(\epsilon)$ then $\left\|\mathscr{H}(\theta^1, t_1) - \mathscr{H}(\theta^2, t_2)\right\| < \epsilon$. In order to construct $\delta(\epsilon)$, though, we first need to establish several other concepts. We begin by examining σ^j_ϕ for each $j \in \{1, \ldots, \ell\}$ and $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. Recognize that for each $j \in \{1, \ldots, \ell\}$ and $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, σ^j_ϕ is formed as the intersection of d linearly independent equalities. Also, recall from Property 5.38 that the defining inequalities of Ω are polynomial in ϕ and linear in σ. Hence, for each $j \in \{1, \ldots, \ell\}$ there is a system of equations of the form $A_j(\phi)\sigma = b_j(\phi)$ for which: (i) the elements of $A_j(\phi)$ and $b_j(\phi)$ are polynomial in ϕ, (ii) $A_j(\phi)$ is invertible for all $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, and (iii) for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, the solution to the system is σ^j_ϕ. This shows that for each $j \in \{1, \ldots, \ell\}$, σ^j_ϕ can be viewed as a continuous function of ϕ. Explicitly, we have:

$$\sigma^j_{(\cdot)} : \Phi^{(d)}_{\Omega,\mathcal{S}} \to \Omega^\sigma \quad \text{such that} \quad \sigma^j_\phi = \left(A_j(\phi)\right)^{-1} b_j(\phi). \tag{5.43}$$

This implies that for all $j \in \{1, \ldots, \ell\}$ and $\epsilon > 0$ there exists $\eta^j_\epsilon > 0$ such that, if $\left\|\phi^1 - \phi^2\right\| \leq \eta^j_\epsilon$ then $\left\|\sigma^j_{\phi^1} - \sigma^j_{\phi^2}\right\| \leq \epsilon$. For each $\epsilon > 0$ define $\delta^1(\epsilon) := \min\left\{\delta : \delta = \eta^j_\epsilon, j \in \{1, \ldots, \ell\}\right\}$. Then we have:

$$\text{For all } \epsilon > 0, \text{ if } \left\|\phi^1 - \phi^2\right\| \leq \delta^1(\epsilon) \text{ then } \left\|\sigma^j_{\phi^1} - \sigma^j_{\phi^2}\right\| \leq \epsilon \text{ for all } j \in \{1, \ldots, \ell\}. \tag{5.44}$$

Now consider the mapping $\psi_\phi(\cdot)$ for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$. Since ψ_ϕ is continuous for each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ we have that for all $\epsilon > 0$ and $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ there exists $\beta^\phi_\epsilon > 0$ such that, if $\|t_1 - t_2\| \leq \beta^\phi_\epsilon$ then

$\|\psi_\phi(t_1) - \psi_\phi(t_2)\| \leq \epsilon$. For each $\epsilon > 0$ define $\delta^2(\epsilon) := \min\left\{\delta : \delta = \beta_\epsilon^\phi \text{ for some } \phi \in \Phi_{\Omega,\mathcal{S}}^{(d)}\right\}$. Then we have that:

$$\text{For all } \epsilon > 0, \text{ if } \|t_1 - t_2\| \leq \delta^2(\epsilon) \text{ then } \|\psi_\phi(t_1) - \psi_\phi(t_2)\| \leq \epsilon \text{ for all } \phi \in \Phi_{\Omega,\mathcal{S}}^{(d)}. \tag{5.45}$$

Next, recognize that the mapping $\psi_\phi(\cdot)$ can be defined in so that, for all $\epsilon > 0$ there exists $\delta^3(\epsilon)$ such that, if $\left\|\phi^1 - \phi^2\right\| \leq \delta^3(\epsilon)$ then $\left\|\psi_{\phi^1}(t) - \psi_{\phi^2}(t)\right\| \leq \epsilon$ for all $t \in [0,1]$. To see this, given $\epsilon > 0$ and distinct $\phi', \phi'' \in \Phi_{\Omega,\mathcal{S}}^{(d)}$, define

$$\gamma_\epsilon^{\phi',\phi''} := \max\left\{\gamma : \gamma \leq \epsilon, \phi'' \in B_\gamma(\phi'), B_\gamma(\phi') \cap \Phi_{\Omega,\mathcal{S}}^{(d)} \text{ is semi-algebraically connected}\right\}.$$

Then for each pair $(\phi^1, \phi^2) \in \Phi_{\Omega,\mathcal{S}}^{(d)}$ define the mapping $\omega_{\phi^1,\phi^2}(\cdot) : [0,1] \to \left(B_{\gamma_\epsilon^{\phi^1,\phi^2}}(\phi^1) \cap \Phi_{\Omega,\mathcal{S}}^{(d)}\right)$ which denotes the continuous semi-algebraic mapping with $\omega(0) = \phi^1$ and $\omega(1) = \phi^2$, as outlined in Definition 5.41. Now select any $\zeta \in (0,1)$ and recognize that given any pair $(\phi^1, \phi^2) \in \Phi_{\Omega,\mathcal{S}}^{(d)}$ for which $\left\|\phi^1 - \phi^2\right\| \leq \gamma_\epsilon^{\phi^1,\phi^2}$, we can define $\psi_{\phi^1}^*(t) := \begin{cases} \phi^1 & \text{if } t < \zeta \\ \psi_{\phi^1}\left(\frac{t-\zeta}{1-\zeta}\right) & \text{otherwise} \end{cases}$ and $\psi_{\phi^2}^*(t) := \begin{cases} \omega_{\phi^1,\phi^2}\left(\frac{t}{\zeta}\right) & \text{if } t < \zeta \\ \psi_{\phi^1}\left(\frac{t-\zeta}{1-\zeta}\right) & \text{otherwise} \end{cases}$. Recognize that each of these mappings satisfy the requirements of Definition 5.41 and moreover, they ensure that $\left\|\psi_{\phi^1}^*(t) - \psi_{\phi^2}^*(t)\right\| \leq \epsilon$ for all $t \in [0,1]$. Hence, if for every $\epsilon > 0$ we define

$$\delta^3(\epsilon) := \min\left\{\delta : \delta = \gamma_\epsilon^{\phi',\phi''} \text{ for some } \phi' \in \Phi_{\Omega,\mathcal{S}}^{(d)} \text{ and } \phi'' \in \Phi_{\Omega,\mathcal{S}}^{(d)}\right\},$$

we obtain the desired result:

$$\text{For all } \epsilon > 0, \text{ if } \left\|\phi^1 - \phi^2\right\| \leq \delta^3(\epsilon) \text{ then } \left\|\psi_{\phi^1}(t) - \psi_{\phi^2}(t)\right\| \leq \epsilon \text{ for all } t \in [0,1]. \tag{5.46}$$

We now show that for any $\epsilon > 0$ there exists $\delta^4(\epsilon)$ such that if $\left\|(\phi^1, \sigma^1) - (\phi^2, \sigma^2)\right\| < \delta^4(\epsilon)$ then $\left|\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2}\right| < \epsilon$ for all $j \in \{1, \ldots, \ell\}$. Recognize that $\left\|(\phi^1, \sigma^1) - (\phi^2, \sigma^2)\right\| < \delta^4(\epsilon)$ implies

$$\left\|\phi^1 - \phi^2\right\| < \delta^4(\epsilon) \tag{5.47}$$

and

$$\left\|\sigma^1 - \sigma^2\right\| < \delta^4(\epsilon). \tag{5.48}$$

We select $\delta^4(\epsilon)$ so that the following hold:

$$\delta^4(\epsilon) < \frac{\epsilon}{2n\ell} \tag{5.49}$$

$$\delta^4(\epsilon) < \delta^1\left(\frac{\epsilon}{2n\ell}\right) \tag{5.50}$$

Thus, from (5.44) we have $\left\|\sigma^j_{\phi^1} - \sigma^j_{\phi^2}\right\| \leq \frac{\epsilon}{2n\ell}$ for all $j \in \{1, \ldots, \ell\}$. This shows that there exists a vector $\alpha \in \mathbb{R}^{k-p}$ such that $\sigma^j_{\phi^2} = \sigma^j_{\phi^1} + \alpha$ and $|\alpha_i| < \frac{\epsilon}{2n\ell}$ for all $i \in \{1, \ldots, k-p\}$. Then from (5.48) we have:

$$\begin{aligned}
\delta^4(\epsilon) &> \left\|\sigma^1 - \sigma^2\right\| \\
&= \left\|\sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\phi^1} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma^j_{\phi^2}\right\| \\
&= \left\|\sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\phi^1} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \left(\sigma^j_{\phi^1} + \alpha\right)\right\| \\
&= \left\|\sum_{j=1}^{\ell} \left(\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2}\right) \sigma^j_{\phi^1} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \alpha\right\| \\
&\geq \left\|\sum_{j=1}^{\ell} \left(\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2}\right) \sigma^j_{\phi^1}\right\| - \left\|\sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \alpha\right\|
\end{aligned}$$

$$\begin{aligned}
\Rightarrow \left\|\sum_{j=1}^{\ell} \left(\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2}\right) \sigma^j_{\phi^1}\right\| &< \delta^4(\epsilon) + \left\|\sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \alpha\right\| \\
&\leq \delta^4(\epsilon) + \sum_{j=1}^{\ell} \left\|\mu_j^{\phi^2,\sigma^2} \alpha\right\| \\
&= \delta^4(\epsilon) + \sum_{j=1}^{\ell} \left|\mu_j^{\phi^2,\sigma^2}\right| \|\alpha\| \\
&= \delta^4(\epsilon) + \|\alpha\| \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \\
&\leq \delta^4(\epsilon) + \|\alpha\| \\
&< \delta^4(\epsilon) + \frac{\epsilon}{2n\ell} \\
&< \frac{\epsilon}{n\ell}
\end{aligned} \tag{5.51}$$

Let $\sigma^* = \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma_{\phi^1}^j$ and recognize that $\sigma^1, \sigma^* \in \Omega^\sigma(\phi^1)$ and furthermore,

$$\mu_j^{\phi^1,\sigma^*} = \mu_j^{\phi^2,\sigma^2} \text{ for each } j \in \{1, \ldots, \ell\} \tag{5.52}$$

since, by construction, the multipliers $\mu^{\phi,\sigma}$ are unique for each $(\phi, \sigma) \in \Lambda$. Also recognize that we have now shown that

$$\left\|\sigma^1 - \sigma^2\right\| < \delta^4(\epsilon) \Rightarrow \left\|\sigma^1 - \sigma^*\right\| < \frac{\epsilon}{n\ell}. \tag{5.53}$$

We now consider arbitrary σ', σ'' such that: (i) there exists $\phi \in \Phi_{\Omega,\mathcal{S}}^{(d)}$ for which $\sigma', \sigma'' \in \Omega^\sigma(\phi)$, and (ii) $\|\sigma' - \sigma''\| < \frac{\epsilon}{n\ell}$. Consider the following two cases: (i) σ' and σ'' lie within the same simplex of $\Omega^\sigma(\phi)$, i.e., there exists $i \in \{1, \ldots, n\}$ such that $\sigma', \sigma'' \in \Delta_i(\phi)$, or (ii) σ' and σ'' lie within different simplices of $\Omega^\sigma(\phi)$, i.e., there is no $i \in \{1, \ldots, n\}$ for which σ' and σ'' are both contained in $\Delta_i(\phi)$. We begin with the former case. Without loss of generality, assume the simplex containing both σ' and σ'' is generated using the first $d+1$ of the ℓ extreme points of $\Omega^\sigma(\phi)$. Hence, we have

$$\mu_j^{\phi,\sigma'} = \mu_j^{\phi,\sigma''} = 0 \text{ for all } j \in \{d+2, \ldots, \ell\} \tag{5.54}$$

and thus

$$\left\|\sum_{j=1}^{d+1} \left(\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma''}\right) \sigma_\phi^j\right\| < \frac{\epsilon}{n\ell}. \tag{5.55}$$

Now recognize that because $\sigma_\phi^1, \ldots, \sigma_\phi^{d+1}$ are affinely independent, we can assume without loss of generality that σ_ϕ^{d+1} is the zero vector and for each $j \in \{1, \ldots, d\}$, σ_ϕ^j contains all zeros, except for a one in the j^{th} position. If this was not the case, it could be achieved by a substitution of parameters, which we now explain. Let $E \in \mathbb{R}^{(k-p)\times d}$ be the matrix whose columns are given by $\sigma_\phi^1 - \sigma_\phi^{d+1}, \ldots, \sigma_\phi^d - \sigma_\phi^{d+1}$. Recognize that, by construction, the columns of E are linearly independent, and thus, there exists an invertible matrix $\tilde{E}$ formed as the product of elementary matrices and permutation matrices, for which the product $\tilde{E}E$ is in reduced row echelon form, i.e., $\tilde{E}E = \begin{bmatrix} I \\ 0 \end{bmatrix}$. Hence, we can obtain an equivalent mpLCP with the property that σ_ϕ^{d+1} is the zero vector and for each $j \in \{1, \ldots, d\}$, σ_ϕ^j contains all zeros, except for a one in the j^{th} position, by replacing σ with $\tilde{E}\left(\sigma - \sigma_\phi^{d+1}\right)$. Now, under the assumption that $\sigma_{\phi^1}^{d+1}$ is the zero vector and for each $j \in \{1, \ldots, d\}$,

σ_ϕ^j contains all zeros, except for a one in the j^{th} position, (5.54) and (5.55) show that for each $j \in \{1, \ldots, d, d+2, \ldots, \ell\}$ we have

$$\left|\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma''}\right| < \frac{\epsilon}{n\ell}. \tag{5.56}$$

Also recognize that we have

$$\begin{aligned}\left|\mu_{d+1}^{\phi,\sigma'} - \mu_{d+1}^{\phi,\sigma''}\right| &= \left|\left(1 - \sum_{j \neq d+1} \mu_j^{\phi,\sigma'}\right) - \left(1 - \sum_{j \neq d+1} \mu_j^{\phi,\sigma''}\right)\right| \\ &= \left|\sum_{j \neq d+1} \left(\mu_j^{\phi,\sigma''} - \mu_j^{\phi,\sigma'}\right)\right| \\ &\leq \sum_{j \neq d+1} \left|\left(\mu_j^{\phi,\sigma''} - \mu_j^{\phi,\sigma'}\right)\right| \\ &< \ell \frac{\epsilon}{n\ell} \\ &= \frac{\epsilon}{n}. \end{aligned} \tag{5.57}$$

This shows that the following result holds:

$$\begin{gathered}\text{Given } \phi \in \Phi_{\Omega,\mathcal{S}}^{(d)} \text{ and } \sigma', \sigma'' \in \Omega^\sigma(\phi) \text{ such that } \|\sigma' - \sigma''\| < \frac{\epsilon}{n\ell}, \text{ if there exists} \\ i \in \{1, \ldots, n\} \text{ such that } \sigma', \sigma'' \in \Delta_i(\phi) \text{ then} \left|\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma''}\right| < \frac{\epsilon}{n} \text{ for all } j \in \{1, \ldots, \ell\}. \end{gathered} \tag{5.58}$$

Now, suppose that σ' and σ'' do not lie within the same simplex of $\Omega^\sigma(\phi)$. Consider the mapping

$$\xi(\cdot) : [0,1] \to \Omega^\sigma(\phi) \text{ such that } \xi(t) = (1-t)\sigma' + t\sigma'' \tag{5.59}$$

and the corresponding set

$$\mathscr{M} := \{\sigma : \exists t \in [0,1] \text{ s.t. } \sigma = \xi(t)\}. \tag{5.60}$$

For each $i \in \{1, \ldots, n\}$, consider the set

$$\mathscr{A}_i := \begin{cases} \underset{\sigma \in (\Delta_i(\phi) \cap \mathscr{M})}{\operatorname{argmax}} \|\sigma' - \sigma\| & \text{if } \Delta_i(\phi) \cap \mathscr{M} \neq \emptyset \\ \emptyset & \text{otherwise} \end{cases}. \tag{5.61}$$

We then define

$$\mathscr{A} := \bigcup_{i \in \{1,\ldots,n\}} \mathscr{A}_i. \tag{5.62}$$

Assume $|\mathscr{A}| = m$ and recognize from (5.61) and (5.62) that $\mathscr{A}$ contains at most one element for each $i \in \{1,\ldots,n\}$. Thus, we have $m \leq n$. Further assume that the elements of $\mathscr{A}$ are ordered as $\sigma^{(1)},\ldots,\sigma^{(m)}$, where $\sigma^{(i)} = \xi(t_i)$ for all $i \in \{1,\ldots,m\}$ and $t_1 \leq \cdots \leq t_m$. Recognize that by construction $\sigma^{(m)} = \sigma''$. Further recognize that because $\xi(\cdot)$ is continuous, for each $i \in \{1,\ldots,m-1\}$ there must exist $j \in \{1,\ldots,n\}$ such that $\sigma^{(i)},\sigma^{(i+1)} \in \Delta_j(\phi)$. Additionally, there must exist $j \in \{1,\ldots,n\}$ such that $\sigma' \in \Delta_j(\phi)$ and $\sigma^{(1)} \in \Delta_j(\phi)$. Moreover, by construction, the graph of $\xi(\cdot)$ is a line segment, and therefore for any pair (σ^i,σ^j) of elements in $\{\sigma',\sigma'',\sigma^{(1)},\ldots,\sigma^{(m)}\}$ we have $\left\|\sigma^i - \sigma^j\right\| \leq \|\sigma' - \sigma''\| < \frac{\epsilon}{n\ell}$. Thus, from (5.58), for each $j \in \{1,\ldots,\ell\}$ we obtain

$$\begin{aligned}
\left|\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma''}\right| &= \left|\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma^{(m)}}\right| \\
&= \left|\mu_j^{\phi,\sigma'} + \sum_{i=1}^{m-1}\left(-\mu_j^{\phi,\sigma^{(i)}} + \mu_j^{\phi,\sigma^{(i)}}\right) - \mu_j^{\phi,\sigma^{(m)}}\right| \\
&\leq \left|\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma^{(1)}}\right| + \sum_{i=1}^{m-1}\left|\mu_j^{\phi,\sigma^{(i)}} - \mu_j^{\phi,\sigma^{(i+1)}}\right| \\
&< m\frac{\epsilon}{n} \\
&\leq \epsilon.
\end{aligned} \tag{5.63}$$

From the results of these two cases, (5.58) when σ' and σ'' lie within the same simplex of $\Omega^\sigma(\phi)$ and (5.63) when σ' and σ'' do not lie within the same simplex of $\Omega^\sigma(\phi)$, we make the following conclusion:

$$\begin{aligned}
&\text{Given } \phi \in \Phi_{\Omega,\mathcal{S}}^{(d)} \text{ and } \sigma',\sigma'' \in \Omega^\sigma(\phi) \text{ such that } \|\sigma' - \sigma''\| < \frac{\epsilon}{n\ell}, \\
&\qquad\qquad \left|\mu_j^{\phi,\sigma'} - \mu_j^{\phi,\sigma''}\right| < \epsilon \text{ for all } j \in \{1,\ldots,\ell\}.
\end{aligned} \tag{5.64}$$

Recognize that the following holds:

$$\begin{aligned}
\left\|(\phi^1,\sigma^1) - (\phi^2 - \sigma^2)\right\| < \delta^4(\epsilon) &\Rightarrow \left\|\sigma^1 - \sigma^2\right\| < \delta^4(\epsilon) && \text{(From (5.48))} \\
&\Rightarrow \left\|\sigma^1 - \sigma^*\right\| < \frac{\epsilon}{n\ell} && \text{(From (5.53))}
\end{aligned}$$

$$\Rightarrow \left|\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^1,\sigma^*}\right| < \epsilon \text{ for all } j \in \{1,\ldots,\ell\} \qquad \text{(From (5.64))}$$
$$\Rightarrow \left|\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2}\right| < \epsilon \text{ for all } j \in \{1,\ldots,\ell\} \qquad \text{(From (5.52))}$$

Hence, we have now shown the following:

$$\text{For all } \epsilon > 0, \text{ if } \left\|(\phi^1,\sigma^1) - (\phi^2,\sigma^2)\right\| < \delta^4(\epsilon) \text{ then } \left|\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2}\right| < \epsilon \text{ for all } j \in \{1,\ldots,\ell\}. \tag{5.65}$$

For each $\epsilon > 0$ we now define

$$\delta(\epsilon) := \min\left\{\epsilon, \delta^2\left(\delta^1\left(\frac{\epsilon}{3\ell}\right)\right), \delta^3\left(\delta^1\left(\frac{\epsilon}{3\ell}\right)\right), \delta^4\left(\frac{\epsilon}{3\ell \max_j\left\{\left\|\sigma^j_{\psi_{\phi^2}(t_2)}\right\|\right\}}\right)\right\}, \tag{5.66}$$

which we use to establish the continuity of $\mathscr{H}$. Recognize in (5.66) that $\max_j\left\{\left\|\sigma^j_{\psi_{\phi^2}(t_2)}\right\|\right\}$ exists because S_θ is bounded. Let pairs (θ^1, t_1) and (θ^2, t_2) be given such that $\theta^1, \theta^2 \in \Lambda$, $t_1, t_2 \in [0,1]$, and $\left\|(\theta^1,t_1) - (\theta^2,t_2)\right\| \le \delta(\epsilon)$. Recall from (5.42) that $\left\|(\theta^1,t_1) - (\theta^2,t_2)\right\| \le \delta(\epsilon)$ if and only if there exist $\phi^1, \phi^2 \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, $\sigma^1 \in \Omega^\sigma(\phi^1)$, and $\sigma^2 \in \Omega^\sigma(\phi^2)$ such that $\theta^1 = (\phi^1,\sigma^1)$ and $\theta^2 = (\phi^2,\sigma^2)$ and the following also hold:

$$\left\|\phi^1 - \phi^2\right\| \le \delta(\epsilon) \tag{5.67}$$

$$\left\|\sigma^1 - \sigma^2\right\| \le \delta(\epsilon) \tag{5.68}$$

and

$$\|t_1 - t_2\| \le \delta(\epsilon). \tag{5.69}$$

Notice that (5.66), (5.69), (5.45) and (5.44) imply that

$$\left\|\sigma^j_{\psi_{\phi^1}(t_1)} - \sigma^j_{\psi_{\phi^1}(t_2)}\right\| \le \frac{\epsilon}{3\ell} \text{ for all } j \in \{1,\ldots,\ell\}, \tag{5.70}$$

and (5.66), (5.67), (5.46) and (5.44) imply that

$$\left\|\sigma^j_{\psi_{\phi^1}(t_2)} - \sigma^j_{\psi_{\phi^2}(t_2)}\right\| \le \frac{\epsilon}{3\ell} \text{ for all } j \in \{1,\ldots,\ell\}. \tag{5.71}$$

Now recognize that

$$\begin{aligned}\left\|\mathscr{H}(\theta^1,t_1)-\mathscr{H}(\theta^2,t_2)\right\| &= \left\|\left(\phi^1,\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_1)}\right)-\left(\phi^2,\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right)\right\| \\ &= \max\left\{\left\|\phi^1-\phi^2\right\|,\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_1)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\|\right\} \qquad (5.72)\end{aligned}$$

Finally, observe the following:

$$\begin{aligned}&\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_1)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &= \left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_1)}-\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}+\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &\le \left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_1)}-\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}\right\|+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &= \left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\left(\sigma^j_{\psi_{\phi^1}(t_1)}-\sigma^j_{\psi_{\phi^1}(t_2)}\right)\right\|+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &\le \sum_{j=1}^{\ell}\left(\left|\mu_j^{\phi^1,\sigma^1}\right|\left\|\sigma^j_{\psi_{\phi^1}(t_1)}-\sigma^j_{\psi_{\phi^1}(t_2)}\right\|\right)+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &\le \sum_{j=1}^{\ell}\left\|\sigma^j_{\psi_{\phi^1}(t_1)}-\sigma^j_{\psi_{\phi^1}(t_2)}\right\|+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \qquad (5.73)\end{aligned}$$

From (5.66) and (5.70), (5.73) implies:

$$\begin{aligned}&\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_1)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &< \ell\frac{\epsilon}{3\ell}+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &= \frac{\epsilon}{3}+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^2}(t_2)}+\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^2}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &\le \frac{\epsilon}{3}+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^1}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^2}(t_2)}\right\|+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^2}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\| \\ &= \frac{\epsilon}{3}+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\left(\sigma^j_{\psi_{\phi^1}(t_2)}-\sigma^j_{\psi_{\phi^2}(t_2)}\right)\right\|+\left\|\sum_{j=1}^{\ell}\mu_j^{\phi^1,\sigma^1}\sigma^j_{\psi_{\phi^2}(t_2)}-\sum_{j=1}^{\ell}\mu_j^{\phi^2,\sigma^2}\sigma^j_{\psi_{\phi^2}(t_2)}\right\|\end{aligned}$$

$$\leq \frac{\epsilon}{3} + \sum_{j=1}^{\ell} \left(\left| \mu_j^{\phi^1,\sigma^1} \right| \left\| \sigma^j_{\psi_{\phi^1}(t_2)} - \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \right) + \left\| \sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\psi_{\phi^2}(t_2)} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma^j_{\psi_{\phi^2}(t_2)} \right\|$$
$$\leq \frac{\epsilon}{3} + \sum_{j=1}^{\ell} \left\| \sigma^j_{\psi_{\phi^1}(t_2)} - \sigma^j_{\psi_{\phi^2}(t_2)} \right\| + \left\| \sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\psi_{\phi^2}(t_2)} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \tag{5.74}$$

Now, from (5.66) and (5.71), (5.74) gives:

$$\begin{aligned} \left\| \sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\psi_{\phi^1}(t_1)} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma^j_{\psi_{\phi^2}(t_2)} \right\| &< \frac{\epsilon}{3} + \ell \frac{\epsilon}{3\ell} + \left\| \sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\psi_{\phi^2}(t_2)} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \\ &= \frac{2\epsilon}{3} + \left\| \sum_{j=1}^{\ell} \left(\mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2} \right) \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \\ &\leq \frac{2\epsilon}{3} + \sum_{j=1}^{\ell} \left(\left| \mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2} \right| \left\| \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \right) \\ &\leq \frac{2\epsilon}{3} + \max_j \left\| \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \left(\sum_{j=1}^{\ell} \left| \mu_j^{\phi^1,\sigma^1} - \mu_j^{\phi^2,\sigma^2} \right| \right) \end{aligned} \tag{5.75}$$

From (5.65) and (5.66), we now see that (5.75) implies:

$$\begin{aligned} \left\| \sum_{j=1}^{\ell} \mu_j^{\phi^1,\sigma^1} \sigma^j_{\psi_{\phi^1}(t_1)} - \sum_{j=1}^{\ell} \mu_j^{\phi^2,\sigma^2} \sigma^j_{\psi_{\phi^2}(t_2)} \right\| &< \frac{2\epsilon}{3} + \ell \max_j \left\| \sigma^j_{\psi_{\phi^2}(t_2)} \right\| \frac{\epsilon}{3\ell \max_j \left\| \sigma^j_{\psi_{\phi^2}(t_2)} \right\|} \\ &= \frac{2\epsilon}{3} + \frac{\epsilon}{3} \\ &= \epsilon \end{aligned} \tag{5.76}$$

Recognize that (5.66) and (5.67) show that $\left\| \phi^1 - \phi^2 \right\| < \epsilon$. Therefore, from this, (5.76) and (5.72) we conclude that $\left\| \mathscr{H}(\theta^1, t_1) - \mathscr{H}(\theta^2, t_2) \right\| < \epsilon$ and thus, we have now shown that $\mathscr{H}$ is continuous.

$\mathscr{H}(\cdot, t)$ is a homeomorphism for each fixed $t \in [0, 1]$: Assume for the arguments that follow that t is fixed. Clearly, $\mathscr{H}(\cdot, t)$ is continuous since $\mathscr{H}$ is continuous. Recognize that $\mathscr{H}(\cdot, t)$ maps each $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$ to itself and each $\sigma \in \Omega^\sigma(\phi)$ to a unique element of $\Omega^\sigma(\psi_\phi(t))$ (due to the method used for selecting $\mu^{\phi,\sigma}$). Moreover, since $\Omega^\sigma(\psi_\phi(t))$ is partitioned in the same fashion as $\Omega^\sigma(\phi)$, every element of $\Omega^\sigma(\psi_\phi(t))$ is mapped to by a unique $\sigma \in \Omega^\sigma(\phi)$ under $\mathscr{H}(\cdot, t)$. Hence, $\mathscr{H}(\cdot, t)$ is bijective. The final property we must show in order to establish that $\mathscr{H}(\cdot, t)$ is a homeomorphism, is that

$\mathscr{H}(\cdot,t)$ has a continuous inverse. We denote this inverse as $\mathscr{H}_t^{-1}(\cdot)$. Explicitly, the inverse can be computed as:

$$\begin{aligned}\mathscr{H}_t^{-1}(\theta) &= \mathscr{H}_t^{-1}(\phi,\sigma)\\ &= \mathscr{H}_t^{-1}\left(\phi, \sum_{j=1}^{\ell} \mu_j^{\phi,\sigma}\sigma^j_{\psi_\phi(t)}\right)\\ &= \left(\phi, \sum_{j=1}^{\ell} \mu_j^{\phi,\sigma}\sigma^j_{\phi}\right)\end{aligned}$$

Clearly the structure of $\mathscr{H}_t^{-1}(\cdot)$ is analogous to that of $\mathscr{H}(\cdot,t)$ and thus, analogous arguments to those used to show the continuity of $\mathscr{H}$ can be used to show the continuity of $\mathscr{H}_t^{-1}(\cdot)$. Hence, we have that $\mathscr{H}(\cdot,t)$ is a homeomorphism.

$\underline{\mathscr{H}(\mathscr{T}^{(d)}_{\Omega,\mathcal{S}},0) = \mathscr{T}^{(d)}_{\Omega,\mathcal{S}}}$: This is clear from (5.37) and the fact that $\psi_\phi(0) = \phi$.

$\underline{\mathscr{H}(\mathscr{T}^{(d)}_{\Omega,\mathcal{S}},1) = \mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi^*)}$: This is established from (5.38) and the facts that $\psi_\phi(1) = \phi^*$ and for a given $t \in [0,1]$, $\mathscr{H}(\mathscr{T}^{(d)}_{\Omega,\mathcal{S}},t)$ maps every ϕ to itself and every possible convex combination of the extreme points of $\Omega^\sigma(\phi)$ to every possible convex combination of the extreme points of $\Omega^\sigma(\psi_\phi(t))$.

As we have now shown that $\mathscr{H}$ satisfies all properties of Definition 5.39, we have that $\mathscr{T}^{(d)}_{\Omega,\mathcal{S}}$ is ambient isotopic to $\mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi^*)$. □

Proposition 5.48. Given $\Omega \subset S_\theta$ having Property 5.38, $dim(\Omega) = \max_{d\in\mathbb{N},\mathcal{S}\in\mathscr{D}_\Omega} \left\{dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) + d\right\}$.

Proof. Recognize that for every $d \in \mathbb{N}$ and every $\mathcal{S} \in \mathscr{D}_\Omega$, $\mathscr{T}^{(d)}_{\Omega,\mathcal{S}}$ is a semi-algebraic subset of Ω. Hence, for all $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_\Omega$ we have:

$$\begin{aligned}dim(\Omega) &\geq dim(\mathscr{T}^{(d)}_{\Omega,\mathcal{S}})\\ &= dim(\mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi)) \quad \text{for any } \phi \in \Phi^{(d)}_{\Omega,\mathcal{S}} \qquad \text{(By Proposition 5.47)}\end{aligned}$$

Notice from (5.38) that for every $\phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}$, $\mathscr{U}^{(d)}_{\Omega,\mathcal{S}}(\phi) = \Phi^{(d)}_{\Omega,\mathcal{S}} \times \Omega^\sigma(\phi)$ and thus for each $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_\Omega$ we have:

$$\begin{aligned}dim(\Omega) &\geq dim(\Phi^{(d)}_{\Omega,\mathcal{S}} \times \Omega^\sigma(\phi)) && \text{for all } \phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}\\ &= dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) + dim(\Omega^\sigma(\phi)) && \text{for all } \phi \in \Phi^{(d)}_{\Omega,\mathcal{S}}\\ &= dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) + d\end{aligned}$$

Since Ω is a bounded set in S_θ, there must be a $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_\Omega$ for which $dim(\Phi^{(d)}_{\Omega,\mathcal{S}})+d$ is maximal. Furthermore, we see from Observation 5.43 that $dim(\Omega)$ must equal $\max_{d\in\mathbb{N},\mathcal{S}\in\mathscr{D}_\Omega} \Big\{dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) + d\Big\}$. □

We have now developed the theory which will allow us to establish necessary and sufficient conditions for any set satisfying Property 5.38 to have dimension k or $k-1$. Consider the following two propositions.

Proposition 5.49. Let $\Omega \subset S_\theta$ satisfying Property 5.38 be given. Then Ω is k-dimensional if and only if there exists $\Phi \subseteq \Omega^\phi$ such that $dim(\Phi) = p$ and $dim\left(\Omega^\sigma(\phi)\right) = k - p$ for all $\phi \in \Phi$.

Proof.
(⇒): Recognize from Proposition 5.48 that since Ω is full dimensional, there must exist $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_\Omega$ such that $k = dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) + d$. Further recognize that since for every $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_\Omega$ we have $dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) \le p$, the following must hold: (i) $d = k - p$, and (ii) $dim(\Phi^{(k-p)}_{\Omega,\mathcal{S}}) = p$. Let $\Phi = \Phi^{(k-p)}_{\Omega,\mathcal{S}}$. Then clearly $\Phi \subseteq \Omega^\phi$ and $dim(\Phi) = p$. Furthermore, from (5.32) and (5.36) we have that for each $\phi \in \Phi$, $dim(\Omega^\sigma(\phi)) = k - p$.
(⇐): Observe from (5.32) that because $dim(\Omega^\sigma(\phi)) = k - p$ for all $\phi \in \Phi$, we have $\Phi \subseteq \Phi^{(k-p)}_\Omega$. Thus, since $dim(\Phi) = p$, we have $dim(\Phi^{(k-p)}_\Omega) \ge p$. However, since $\Phi^{(k-p)}_\Omega \subseteq \mathbb{R}^p$ we also have that $dim(\Phi^{(k-p)}_\Omega) \le p$, which shows that $dim(\Phi^{(k-p)}_\Omega) = p$. Now recognize the following:

- $\Phi^{(k-p)}_\Omega = \cup_{\mathcal{S}\in\mathscr{D}_\Omega}\Phi^{(k-p)}_{\Omega,\mathcal{S}}$. – By (5.32), (5.35) and (5.36).
- $\mathscr{D}_\Omega$ is a finite set. – By (5.34).
- For each $\mathcal{S} \in \mathscr{D}_\Omega$ we have that $\Phi^{(k-p)}_{\Omega,\mathcal{S}}$ is semi-algebraic. – By Corollary 5.46.

These three facts, together with Proposition 5.42, show that there must exist $\mathcal{S}' \in \mathscr{D}_\Omega$ for which $dim(\Phi^{(k-p)}_{\Omega,\mathcal{S}'}) = dim(\Phi^{(k-p)}_\Omega)$. Therefore, by Proposition 5.48, we have $dim(\Omega) \ge dim(\Phi^{(k-p)}_{\Omega,\mathcal{S}'}) + k - p = p + k - p = k$. However, this clearly implies that $dim(\Omega) = k$ since $\Omega \subseteq \mathbb{R}^k$. □

Proposition 5.50. Let $\Omega \subset S_\theta$ satisfying Property 5.38 be given for which $dim(\Omega) \neq k$. Then $dim(\Omega) = k - 1$ if and only if there exists $\Phi \subseteq \Omega^\phi$ for which one of the following two conditions holds:

1. $dim(\Phi) = p$ and for each $\phi \in \Phi$, $dim\left(\Omega^\sigma(\phi)\right) = (k - p) - 1$.
2. $dim(\Phi) = p - 1$ and for each $\phi \in \Phi$, $dim\left(\Omega^\sigma(\phi)\right) = k - p$.

Proof.

($\Rightarrow$): Since $dim(\Omega) = k-1$, by Proposition 5.48 there must exist $d \in \mathbb{N}$ and $\mathcal{S} \in \mathscr{D}_\Omega$ such that $dim(\Phi^{(d)}_{\Omega,\mathcal{S}}) + d = k-1$. Recognize that there are now two possible cases: (i) $d = (k-p)-1$ and $dim(\Phi^{(k-p-1)}_{\Omega,\mathcal{S}}) = p$, or (ii) $d = k-p$ and $dim(\Phi^{(k-p)}_{\Omega,\mathcal{S}}) = p-1$. We now examine these cases, one at a time.

Suppose that $d = (k-p)-1$ and $dim(\Phi^{(k-p-1)}_{\Omega,\mathcal{S}}) = p$. Let $\Phi = \Phi^{(k-p-1)}_{\Omega,\mathcal{S}}$. Then clearly $\Phi \subseteq \Omega^\phi$ and $dim(\Phi) = p$. Furthermore, from (5.32) and (5.36) we have that for each $\phi \in \Phi$, $dim(\Omega^\sigma(\phi)) = (k-p)-1$. Hence, condition (1) of the proposition is satisfied.

Next suppose that $d = k-p$ and $dim(\Phi^{(k-p)}_{\Omega,\mathcal{S}}) = p-1$. Let $\Phi = \Phi^{(k-p)}_{\Omega,\mathcal{S}}$. Clearly $\Phi \subseteq \Omega^\phi$ and $dim(\Phi) = p-1$. From (5.32) and (5.36) we have that for each $\phi \in \Phi$, $dim(\Omega^\sigma(\phi)) = k-p$. Hence, condition (2) of the proposition is satisfied.

($\Leftarrow$): Recognize that by utilizing arguments analogous to those used to prove the reverse direction of Proposition 5.49, both conditions (1) and (2) above result in concluding that $dim(\Omega) \geq k-1$. However, since we know that $dim(\Omega) \neq k$, it must be that $\dim(\Omega) \leq k-1$. Thus, $\dim(\Omega) = k-1$. □

Having now established necessary and sufficient conditions for any set satisfying Property 5.38 to have dimension k or $k-1$, we are able to make several important observations about invariancy regions. We discuss these observations in the following subsection.

5.3.3 An Initial Strategy for Partitioning S_θ

In this subsection we provide two corollaries which result directly from the theory developed in Section 5.3.2. We then use the results of these corollaries to prove a proposition which contains the primary theoretical result that ensures the correctness of the methodology we introduce in Sections 5.4 and 5.5. Consider the following two corollaries, the first follows from Proposition 5.49 and the second follows from Proposition 5.50.

Corollary 5.51. Given a feasible complementary basis $\mathcal{B}$, the invariancy region $\mathcal{IR}_\mathcal{B}$ is k-dimensional if and only if there exists $\Phi \subseteq \mathcal{IR}^\phi_\mathcal{B}$ such that $dim(\Phi) = p$ and $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})\right) = k-p$ for all $\phi \in \Phi$.

Proof. The result follows directly from Proposition 5.49, the fact that invariancy regions satisfy Property 5.38, and Proposition 5.36. □

Corollary 5.52. Let a feasible complementary basis $\mathcal{B}$ be given for which $dim(\mathcal{IR}_{\mathcal{B}}) \neq k$. Then $dim(\mathcal{IR}_{\mathcal{B}}) = k-1$ if and only if there exists $\Phi \subseteq \mathcal{IR}^{\phi}_{\mathcal{B}}$ for which one of the following two conditions holds:

1. $dim(\Phi) = p$ and for each $\phi \in \Phi$, $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})\right) = (k-p)-1$.

2. $dim(\Phi) = p-1$ and for each $\phi \in \Phi$, $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})\right) = k-p$.

Proof. The result follows directly from Proposition 5.50, the fact that invariancy regions satisfy Property 5.38, and Proposition 5.36. □

We now use the results of these corollaries and propose another result which will be extremely useful in developing an initial strategy for partitioning S_θ. Consider the following proposition.

Proposition 5.53. Let two feasible complementary bases $\mathcal{B}_i$ and $\mathcal{B}_j$ be given such that $\mathcal{IR}_{\mathcal{B}_i}$ and $\mathcal{IR}_{\mathcal{B}_j}$ are each full dimensional and adjacent, i.e., $dim\left(\mathcal{IR}_{\mathcal{B}_i} \cap \mathcal{IR}_{\mathcal{B}_j}\right) = k-1$. Then there exists a sequence of bases $\{\mathcal{B}_n\}_{n=i+1}^{j-1}$ and $\Phi \subseteq \cap_{n=i}^{j} \mathcal{IR}^{\phi}_{\mathcal{B}_n}$ such that: (i) $dim(\Phi) \geq p-1$, (ii) for each $\phi \in \Phi$, $\mathcal{C}(\phi, \mathcal{B}_\gamma)$ and $\mathcal{C}(\phi, \mathcal{B}_{\gamma+1})$ are adjacent for all $\gamma \in \{i, \ldots, j-1\}$, and (iii) $dim\left(\mathcal{IR}_{\mathcal{B}_\gamma}\right) \geq k-1$ for all $\gamma \in \{i+1, \ldots, j-1\}$.

Proof. Recognize that because $dim\left(\mathcal{IR}_{\mathcal{B}_i} \cap \mathcal{IR}_{\mathcal{B}_j}\right) = k-1$, $dim\left(\mathcal{IR}^{\phi}_{\mathcal{B}_i} \cap \mathcal{IR}^{\phi}_{\mathcal{B}_j}\right)$ is either p or $p-1$. We consider these cases one at a time.

First suppose that $dim\left(\mathcal{IR}^{\phi}_{\mathcal{B}_i} \cap \mathcal{IR}^{\phi}_{\mathcal{B}_j}\right) = p$. Fix any $\phi' \in \left(\mathcal{IR}^{\phi}_{\mathcal{B}_i} \cap \mathcal{IR}^{\phi}_{\mathcal{B}_j}\right)$ and consider the invariancy regions $\mathcal{IR}^{\sigma}_{\mathcal{B}_i}(\phi')$ and $\mathcal{IR}^{\sigma}_{\mathcal{B}_j}(\phi')$. Recognize that with ϕ' fixed, $M(\phi')$ is a real valued matrix, and for any basis $\mathcal{B}$, the cone $\mathcal{C}(\phi', \mathcal{B})$ is simply the conic combination of vectors with real components. Hence, with ϕ' fixed we can consider $\mathcal{IR}^{\sigma}_{\mathcal{B}_i}(\phi')$ and $\mathcal{IR}^{\sigma}_{\mathcal{B}_j}(\phi')$ in the context of the works of [18] and [2], since in these works M is a real valued matrix. Then by Theorem 5.10 of [18], there exists a sequence of invariancy regions $\left\{\mathcal{IR}^{\sigma}_{\mathcal{B}_n}(\phi')\right\}_{n=i+1}^{j-1}$, and by extension, a sequence of bases $\{B_n\}_{n=i+1}^{j-1}$, such that $\mathcal{C}(\phi', \mathcal{B}_\gamma)$ and $\mathcal{C}(\phi', \mathcal{B}_{\gamma+1})$ are adjacent for all $\gamma \in \{i, \ldots, j-1\}$ and $dim\left(\mathcal{AS}(\phi') \cap \mathcal{C}(\phi', \mathcal{B}_\gamma)\right) \geq (k-p)-1$ for all $\gamma \in \{i+1, \ldots, j-1\}$. We say that such a sequence of bases is *valid* for ϕ'. Recognize that valid sequences are not necessarily unique, and furthermore, the same sequence may not be valid for distinct $\phi^*, \phi^{**} \in \left(\mathcal{IR}^{\phi}_{\mathcal{B}_i} \cap \mathcal{IR}^{\phi}_{\mathcal{B}_j}\right)$.

For a given $\phi \in \left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right)$, let $\mathscr{C}^{\phi}$ represent the set of all valid sequences for ϕ and define $\mathscr{C} := \cup_{\phi \in \left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right)} \mathscr{C}^{\phi}$. Then for each $\mathscr{S} \in \mathscr{C}$ define the set

$$\mathscr{V}(\mathscr{S}) := \left\{\phi \in \left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right) : \text{ sequence } \mathscr{S} \text{ is valid for } \phi\right\}. \tag{5.77}$$

Recognize that the following are true:

1. $|\mathscr{C}| < \infty$.

2. $\cup_{\mathscr{S} \in \mathscr{C}} \mathscr{V}(\mathscr{S}) = \left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right)$.

3. For each $\mathscr{S} \in \mathscr{C}$, the set $\mathscr{V}(\mathscr{S})$ is semi-algebraic.

The first is due to the fact that there are a finite number of bases and hence a finite number of sequences. The second is obvious. The third is due to the fact that $\mathscr{V}(\mathscr{S})$ can be represented as

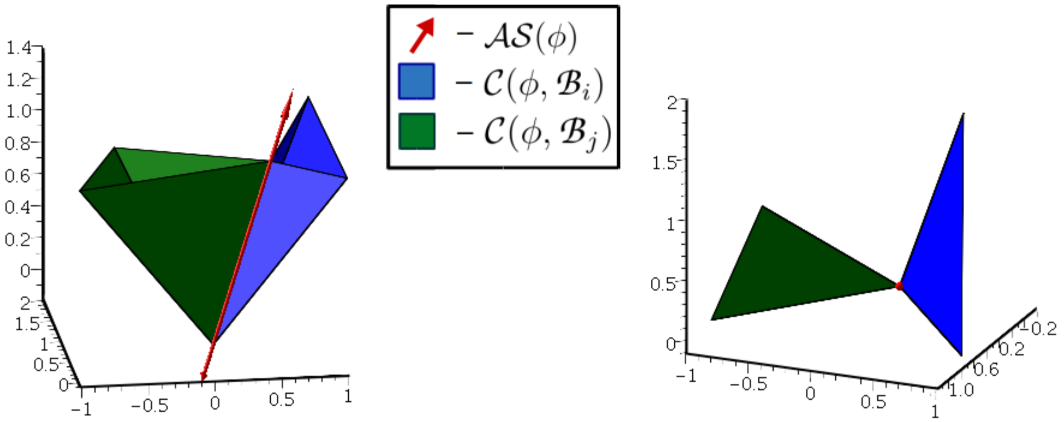
$$\begin{aligned}\mathscr{V}(\mathscr{S}) &= \left(\bigcap_{\mathscr{B} \in \mathscr{S}} \Phi^{(\geq k-p-1)}_{\mathcal{IR}_{\mathscr{B}}}\right) \bigcap \left(\bigcap_{(\mathscr{B}_i, \mathscr{B}_j) \in \mathscr{S}} \{\phi : \mathcal{C}(\phi, \mathscr{B}_i) \text{ is adjacent to } \mathcal{C}(\phi, \mathscr{B}_j)\}\right)\\ &= \left(\bigcap_{\mathscr{B} \in \mathscr{S}} \Phi^{(\geq k-p-1)}_{\mathcal{IR}_{\mathscr{B}}}\right) \bigcap \left(\bigcap_{(\mathscr{B}_i, \mathscr{B}_j) \in \mathscr{S}} \left\{\phi : rank\left(G_{\bullet(\mathscr{B}_i \cap \mathscr{B}_j)}(\phi)\right) = h-1\right\}\right).\end{aligned}$$

Notice that the above set is clearly semi-algebraic if: (i) $\Phi^{(\geq k-p-1)}_{\mathcal{IR}_{\mathscr{B}}}$ is semi-algebraic for each $\mathscr{B} \in \mathscr{S}$, and (ii) the set $\left\{\phi : rank\left(G_{\bullet(\mathscr{B}_i \cap \mathscr{B}_j)}(\phi)\right) = h-1\right\}$ is semi-algebraic for each pair $(\mathscr{B}_i, \mathscr{B}_j) \in \mathscr{S}$. The former is clear from Proposition 5.44 and the fact that invariancy regions satisfy Property 5.38. The arguments needed to show the latter are analogous to those used in the proof of Proposition 5.44 in which we showed that conditions on the rank of a matrix can be imposed using a set of polynomial inequalities. Now, since (1), (2) and (3) hold, we have by Proposition 5.42 that there exists some $\mathscr{S}' \in \mathscr{C}$ for which $dim(\mathscr{V}(\mathscr{S}')) = p$. Hence, if we let $\Phi = \mathscr{V}(\mathscr{S}')$, together $\mathscr{S}'$ and Φ satisfy conditions (i) and (ii) of the proposition. Furthermore, since $dim(\Phi) = p$ and $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathscr{B})\right) \geq (k-p) - 1$ for all $\phi \in \Phi$ and $\mathscr{B} \in \mathscr{S}$, condition (iii) of the proposition is also satisfied by Corollary 5.52.

Now suppose that $dim\left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right) = p-1$. Recognize that $\left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right)$ satisfies Property 5.38 and so by Proposition 5.50 we have that since $dim\left(\mathcal{IR}_{\mathscr{B}_i} \cap \mathcal{IR}_{\mathscr{B}_j}\right) = k-1$, there must exist a $p-1$ dimensional subset Φ' of $\left(\mathcal{IR}^{\phi}_{\mathscr{B}_i} \cap \mathcal{IR}^{\phi}_{\mathscr{B}_j}\right)$ such that $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathscr{B}_i)\right) = k-p$

for all $\phi \in \Phi'$ and $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}_j)\right) = k - p$ for all $\phi \in \Phi'$. Recognize that this can only happen if for each $\phi \in \Phi'$, $\mathcal{C}(\phi, \mathcal{B}_i)$ and $\mathcal{C}(\phi, \mathcal{B}_j)$ share a facet $\mathscr{F}_\phi$ of dimension at least $k - p$ and $\mathcal{AS}(\phi)$ has a $k - p$ dimensional intersection with $\mathscr{F}_\phi$. Recall the following facts: (i) for every feasible complementary basis $\mathcal{B}$ and every $\phi \in S_\phi$, $\mathcal{C}(\phi, \mathcal{B}_\gamma)$ is an h dimensional cone in $\mathbb{R}^h$, and (ii) because $M(\phi)$ is sufficient for each $\phi \in S_\phi$ we have $\mathcal{K}(M(\phi))$ is convex for each $\phi \in S_\phi$ and $\cup_{\text{feasible bases } \mathcal{B}} \mathcal{C}(\phi, \mathcal{B})$ forms a partition of $\mathcal{K}(M(\phi))$ for each $\phi \in S_\phi$. From these three facts, recognize that for every $\phi \in \Phi'$, every $\tau \in relint(\mathscr{F}_\phi \cap \mathcal{AS}(\phi))$ and every $\tau' \in \mathbb{R}^h$, there exists $\epsilon^\phi_{\tau,\tau'} > 0$ such that for all $\epsilon \in (0, \epsilon^\phi_{\tau,\tau'}]$ we have either (i) $\tau + \epsilon\tau' \notin \mathcal{K}(M(\phi))$, or (ii) $\tau + \epsilon\tau' \in \mathcal{K}(M(\phi))$ and the parametric complementary cone containing $\tau + \epsilon\tau'$ also contains $\mathscr{F}_\phi$. For each $\phi \in \Phi'$ define the set $\mathscr{A}(\phi) := \{\mathcal{B} : \exists \tau \in relint(\mathscr{F}_\phi \cap \mathcal{AS}(\phi), \tau' \in \mathbb{R}^h \text{ such that } \tau + \epsilon^\phi_{\tau,\tau'}\tau' \in \mathcal{C}(\phi, \mathcal{B})\}$. Further recognize that due to the convexity of $\mathcal{K}(M(\phi))$ for each $\phi \in S_\phi$, that there must exist a subset $\{\mathcal{B}_1, \ldots, \mathcal{B}_m\}$ of bases in $\mathscr{A}(\phi)$ such that the sequence $\{\mathcal{B}_n\}_{n=1}^m$ satisfies the following properties: (i) $\mathcal{B}_1 = \mathcal{B}_i$, (ii) $\mathcal{B}_m = \mathcal{B}_j$, (iii) $\mathcal{C}(\phi, \mathcal{B}_n)$ and $\mathcal{C}(\phi, \mathcal{B}_{n+1})$ are adjacent for all $n \in \{1, \ldots, m-1\}$, and (iv) and $\mathscr{F}_\phi \subset \mathcal{C}(\phi, \mathcal{B}_n)$ for all $n \in \{1, \ldots, m\}$. As with the case in which $dim\left(\mathcal{IR}^\phi_{\mathcal{B}_i} \cap \mathcal{IR}^\phi_{\mathcal{B}_j}\right) = p$, we say that this sequence is *valid* for ϕ. Hence, as we did in the previous case, for each sequence $\mathscr{S}$ valid for some $\phi \in \Phi'$ we can construct the set $\mathscr{V}(\mathscr{S})$ of $\phi \in \Phi'$ for which $\mathscr{S}$ is valid. Then since we showed that there are a finite number of valid sequences, $\cup_{\mathscr{S} \in \mathscr{C}} \mathscr{V}(\mathscr{S}) = \left(\mathcal{IR}^\phi_{\mathcal{B}_i} \cap \mathcal{IR}^\phi_{\mathcal{B}_j}\right)$, and for each valid sequence $\mathscr{S}$, the set $\mathscr{V}(\mathscr{S})$ is semi-algebraic, we have from Proposition 5.42 that there must exist a valid sequence $\mathscr{S}'$ such that $dim(\mathscr{V}(\mathscr{S}')) = p - 1$. Let $\Phi = \mathscr{V}(\mathscr{S}')$. Then clearly Φ is a $p - 1$ dimensional subset of Φ' such that a single sequence of bases $\mathscr{S}$ is valid for all $\phi \in \Phi$. Hence, conditions (i) and (ii) of the proposition also hold when $dim\left(\mathcal{IR}^\phi_{\mathcal{B}_i} \cap \mathcal{IR}^\phi_{\mathcal{B}_j}\right) = p - 1$. Furthermore, since $dim(\Phi) = p - 1$ and $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})\right) = k - p$ for all $\phi \in \Phi$ and $\mathcal{B} \in \mathscr{S}$, condition (iii) of the proposition is also satisfied by Proposition 5.50. □

To aid in visualization of some of the concepts introduced in the proof of Proposition 5.53, particularly the case in which $dim\left(\mathcal{IR}^\phi_{\mathcal{B}_i} \cap \mathcal{IR}^\phi_{\mathcal{B}_j}\right) = p - 1$, we include Figure 5.1 which displays an example of parametric complementary cones which are not adjacent, but do share a $k-1$ dimensional facet. Note that this example is specific to the special case in which $h = 3$ and $p = k - 1$. Observe, particularly from the top view in Figure 5.1b, that because $\mathcal{K}(M(\phi))$ is convex and partitioned by the set of parametric complementary cones, the missing space "between" cones $\mathcal{C}(\phi, \mathcal{B}_i)$ and $\mathcal{C}(\phi, \mathcal{B}_i)$ must also be partitioned by other parametric complementary cones and furthermore, there must be



(a) Side view

(b) Top view

Figure 5.1: Example of $k - p$ dimensional intersection of $\mathscr{AS}(\phi)$ with two distinct complementary cones. ($h = 3$, $p = k - 1$)

a subset C of these cones such that for each cone in C, the intersection of the cone with $\mathscr{AS}(\phi)$ is $k - 1$ dimensional. Hence, there must be a sequence of cones in C, and by extension a sequence of bases, which satisfy the conditions of Propostion 5.53.

Recognize that Proposition 5.53 indicates that one strategy for partitioning S_θ is to begin with a full dimensional invariancy region, compute the sequences of bases which yield each adjacent full dimensional invariancy region, and then repeat the process for each discovered region until no undiscovered invariancy regions exist within S_θ. In the next section we introduce the theoretical results which we use to carry out these procedures.

5.4 Phase 2: Partitioning the parameter space

In this section we introduce the theory necessary for developing an algorithm that can be used to partition S_θ, given an initial basis $\mathscr{B}_0$ such that $dim\left(\mathcal{IR}_{\mathscr{B}_0}\right) = k$. The algorithm itself is presented at the end of this section. We discuss obtaining this initial basis, and present an algorithm for doing so, in Section 5.5. We note that much of the material in this section is a direct extension of work done in [2]. As a result, the format of this section, up to approximately the introduction of Outline 5.62, replicates the format of Section 3 in [2].

Before we can find sequences of bases which link adjacent invariancy regions as discussed in Proposition 5.53, we must first establish a method so that given a complementary feasible basis $\mathcal{B}$, we can determine a complementary feasible basis $\mathcal{B}'$ which is adjacent to $\mathcal{B}$. Observe the following proposition.

Proposition 5.54. [18] If $M(\phi) \in \mathbb{R}^{h \times h}$ is column sufficient for each $\phi \in S_\phi$ and two bases $\mathcal{B}_1$ and $\mathcal{B}_2$ are adjacent, then $|\mathcal{B}_1 \cap \mathcal{B}_2| \geq h - 2$.

Proof. The result follows by extension of Lemma 3.8 of [18]. □

The result of Proposition 5.54 is quite powerful as it implies that given any basis $\mathcal{B}$, all complementary bases which are adjacent to $\mathcal{B}$ can be obtained by replacing either 1 or 2 elements of $\mathcal{B}$ with their complements.

Definition 5.55. Replacing a single element of a basis with its complement is called a *diagonal pivot.*

Definition 5.56. Replacing two elements of a basis with their complements is called an *exchange pivot.*

These terms arise from techniques for solving LCP which consider the LCP in a tableau format and rely on principal pivoting to find feasible solutions. For a given basis $\mathcal{B}$, the corresponding tableau is the augmented matrix

$$T_{\mathcal{B}}(\phi, \sigma) := \left[\begin{array}{c:c} G(\phi)_{\bullet\mathcal{B}}^{-1} G(\phi) & G(\phi)_{\bullet\mathcal{B}}^{-1} q(\phi, \sigma) \end{array} \right] \tag{5.78}$$

where the right hand side (RHS) is precisely $\nu_{\mathcal{B}}(\theta)$. Note that the elements of the tableau associated with a given basis $\mathcal{B}$ are rational functions of ϕ and σ. However, only the elements of the RHS of the tableau depend on σ. Thus, since the majority of the theory we present in this work for which we utilize tableaux deals with only the LHS elements, when appropriate, we drop the dependency of $T_{\mathcal{B}}$ on σ and use the notation $T_{\mathcal{B}}(\phi)$.

It is important to note that given a basis $\mathcal{B}$, not every diagonal and exchange pivot will result in a new feasible basis. To see this, suppose that in a particular diagonal or exchange pivot $J \subset \mathcal{B}$ is the set of indices replaced with their complements. If $rank\left(G(\phi)_{\bullet\left((\mathcal{B}\setminus J)\cup\overline{J}\right)}\right) \neq h$ for all $\phi \in S_\phi$ then $\left((\mathcal{B} \setminus J) \cup \overline{J}\right)$ cannot be a basis. Additionally, even if a pivot on $\mathcal{B}$ does result in a new basis $\mathcal{B}'$, the bases $\mathcal{B}$ and $\mathcal{B}'$ may not be adjacent. Due to these facts, we next need to determine conditions under which pivots will yield new adjacent bases. Such conditions can be developed using the tableau $T_{\mathcal{B}}(\phi, \sigma)$.

We first consider diagonal pivots. Since principal pivoting has been studied extensively in the context of LCP, the following result is a direct extension of a well known fact in the literature.

Observation 5.57. Given a complementary feasible basis $\mathcal{B}$ and any index $i \in \mathcal{B}$, the set $(\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis if and only if there exists $\phi \in S_\phi$ such that $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} \neq 0$.

The following proposition and its corollary are slightly modified from [18].

Proposition 5.58. Given a complementary feasible basis $\mathcal{B}$, suppose that for some index $i \in \mathcal{B}$ the set $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis. Then $\mathcal{B}$ and $\mathcal{B}'$ are adjacent.

Proof. The proposition is implied by the facts that for all $\phi \in S_\phi$, $\mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}') = cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ and $dim\left(cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)\right) = h - 1$. Therefore, by Definition 5.24, the bases $\mathcal{B}$ and $\mathcal{B}'$ are adjacent. □

Corollary 5.59. Given a complementary feasible basis $\mathcal{B}$, suppose that for some index $i \in \mathcal{B}$ the set $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis. If $M(\phi)$ is column sufficient for all $\phi \in S_\phi$ then for each $\phi \in S_\phi$, $\mathcal{C}(\phi, \mathcal{B}')$ is the unique parametric complementary cone adjacent to $\mathcal{C}(\phi, \mathcal{B})$ along the facet $cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$.

Proof. The Corollary is implied by the fact that for column sufficient matrices the relative interiors of all complementary cones are disjoint. □

Together Observation 5.57 and Proposition 5.58 provide conditions under which an adjacent complementary feasible basis $\mathcal{B}'$ can be derived from a given complementary feasible basis $\mathcal{B}$ by using a single diagonal pivot. We now consider exchange pivots and present the following two new propositions.

Proposition 5.60. For a given complementary basis $\mathcal{B}$, suppose there exist distinct $i, j \in \mathcal{B}$ such that $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a complementary basis which is adjacent to $\mathcal{B}$. Then if $M(\phi)$ is column sufficient for every $\phi \in S_\phi$, either $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} = 0$ for all $\phi \in S_\phi$ or $(T_{\mathcal{B}}(\phi))_{j,\bar{\jmath}} = 0$ for all $\phi \in S_\phi$.

Proof. Assume without loss of generality that there exists $\phi' \in S_\phi$ such that $\left(T_{\mathcal{B}}(\phi')\right)_{i,\bar{\imath}} \neq 0$. Then $\hat{\mathcal{B}} = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a complementary basis and for each $\phi \in S_\phi$, $\mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \hat{\mathcal{B}}) = \mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Furthermore, by Corollary 5.59, $\hat{\mathcal{B}}$ is the unique basis whose parametric complementary cone intersects $\mathcal{C}(\phi, \mathcal{B})$ along this facet for all $\phi \in S_\phi$. Therefore, since $\mathcal{B}'$ is also adjacent to $\mathcal{B}$, it must be that $\mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}') \subseteq \mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right)$ for all $\phi \in S_\phi$. Hence, there must exist $J' \subset \mathcal{B}'$ such that $|J'| = h - 1$ and $\mathit{dim}\left(\mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right) \cap \mathit{cone}\left(G(\phi)_{\bullet J'}\right)\right) = h - 1$ for all $\phi \in S_\phi$. Note that since $|J'| = h - 1$, either $\bar{\imath}$ or $\bar{\jmath}$ must be a member of J'. Consider the following two cases:

Case 1: $\bar{\imath} \in J'$

Since $\mathit{dim}\left(\mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right) \cap \mathit{cone}\left(G(\phi)_{\bullet J'}\right)\right) = h - 1$ for all $\phi \in S_\phi$, we have $G(\phi)_{\bullet\bar{\imath}} \in \mathit{span}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right)$ for all $\phi \in S_\phi$. Now, suppose that $G(\phi)_{\bullet\bar{\imath}} \notin \mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right)$ for all $\phi \in S_\phi$. Then $\mathcal{C}(\phi, \mathcal{B})$ and $\mathcal{C}(\phi, \mathcal{B}')$ can only be adjacent along the facet $\mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right)$ for all $\phi \in S_\phi$ if $\mathcal{C}(\phi, \mathcal{B}) = \mathit{cone}\left(G(\phi)_{\bullet\mathcal{B}}\right) \subset \mathit{cone}\left(G(\phi)_{\bullet\left((\mathcal{B}\setminus\{i\})\cup\{\bar{\imath}\}\right)}\right) = \mathcal{C}(\phi, \hat{\mathcal{B}})$ for all $\phi \in S_\phi$, but this contradicts the fact that $M(\phi)$ is column sufficient for all $\phi \in S_\phi$ since in this case $\mathcal{C}(\phi, \hat{\mathcal{B}})$ and $\mathcal{C}(\phi, \mathcal{B})$ do not have disjoint relative interiors for all $\phi \in S_\phi$. Suppose instead that $G(\phi)_{\bullet\bar{\imath}} \in \mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right)$. This implies that $\mathcal{C}(\phi, \hat{\mathcal{B}}) = \mathit{cone}\left(G(\phi)_{\bullet\left((\mathcal{B}\setminus\{i\})\cup\{\bar{\imath}\}\right)}\right) \subseteq \mathit{cone}\left(G(\phi)_{\bullet\mathcal{B}}\right) = \mathcal{C}(\phi, \mathcal{B})$ for all $\phi \in S_\phi$, which means that one of the following must hold: (i) $\mathcal{C}(\phi, \hat{\mathcal{B}})$ is not full dimensional for all $\phi \in S_\phi$, or (ii) $\mathcal{C}(\phi, \hat{\mathcal{B}})$ and $\mathcal{C}(\phi, \mathcal{B})$ do not have disjoint relative interiors for all $\phi \in S_\phi$. Notice, however, that the former contradicts the fact that $\hat{\mathcal{B}}$ is a basis and the latter contradicts the fact that $M(\phi)$ is column sufficient for all $\phi \in S_\phi$.

Case 2: $\bar{\jmath} \in J'$

Since $\mathit{dim}\left(\mathit{cone}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right) \cap \mathit{cone}\left(G(\phi)_{\bullet J'}\right)\right) = h - 1$ for all $\phi \in S_\phi$, we have $G(\phi)_{\bullet\bar{\jmath}} \in \mathit{span}\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}\right)$ for all $\phi \in S_\phi$. Since $T_{\mathcal{B}}(\phi, \sigma) = \left[\begin{array}{c:c} G(\phi)_{\bullet\mathcal{B}}^{-1} G(\phi) & G(\phi)_{\bullet\mathcal{B}}^{-1} q(\phi, \sigma) \end{array}\right]$, we have $\left(T_{\mathcal{B}}(\phi)\right)_{j,\gamma} = 0$ for all $\gamma \in (\mathcal{B} \setminus \{j\})$. Thus, since for all $\phi \in S_\phi$, $G(\phi)_{\bullet\bar{\jmath}}$ is a linear combination of the columns of $G(\phi)_{\bullet(\mathcal{B}\setminus\{j\})}$, it must be that $\left(T_{\mathcal{B}}(\phi)\right)_{j,\bar{\jmath}} = 0$.

Cases 1 and 2 are now complete. The contradictions found in Case 1 show that either $\bar{\imath} \notin J'$ or $\left(T_{\mathcal{B}}(\phi)\right)_{i,\bar{\imath}} = 0$ for all $\phi \in S_\phi$. Clearly, if $\left(T_{\mathcal{B}}(\phi)\right)_{i,\bar{\imath}} = 0$ for all $\phi \in S_\phi$ the thesis of the proposition holds. If, on the other hand, $\bar{\imath} \notin J'$ then $\bar{\jmath} \in J'$ which, as is shown in Case 2, implies that $\left(T_{\mathcal{B}}(\phi)\right)_{j,\bar{\jmath}} = 0$ for all $\phi \in S_\phi$. Thus, the claim of the proposition holds in all cases. □

Note that aspects of one of the proofs in [18] are used in the proof of the following proposition.

Proposition 5.61. Let a complementary basis $\mathcal{B}$ and distinct $i, j \in \mathcal{B}$ be given. Given $\phi \in \mathcal{IR}^{\phi}_{\mathcal{B}}$, the set $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a complementary basis such that $\mathcal{C}(\phi, \mathcal{B})$ is adjacent to $\mathcal{C}(\phi, \mathcal{B}')$ along the facet *cone* $\left(G(\phi)^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ if and only if $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} = 0$, $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$, and $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} \neq 0$.

Proof. ($\Leftarrow$): We first show that $\mathcal{B}'$ is a basis and then show that $\mathcal{C}(\phi, \mathcal{B})$ is adjacent to $\mathcal{C}(\phi, \mathcal{B}')$ along the facet *cone* $\left(G(\phi)^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Since the LHS of $T_{\mathcal{B}}(\phi)$ is given by $G(\phi)^{-1}_{\bullet\mathcal{B}}G(\phi)$, we have $(T_{\mathcal{B}}(\phi))_{i,\gamma} = 0$ for all $\gamma \in \mathcal{B} \setminus \{i\}$ and $(T_{\mathcal{B}}(\phi))_{j,\xi} = 0$ for all $\xi \in \mathcal{B} \setminus \{j\}$. Thus, since $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$, $G(\phi)_{\bullet\bar{\imath}}$ cannot be a linear combination of the columns of $G(\phi)_{\bullet(\mathcal{B}\setminus\{i,j\})}$. Furthermore, since $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} = 0$ and $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} \neq 0$, $G(\phi)_{\bullet\bar{\jmath}}$ cannot be a linear combination of the columns of $G(\phi)_{\bullet((\mathcal{B}\setminus\{i,j\})\cup\{\bar{\imath}\})}$. Thus, the columns of $G(\phi)_{\bullet\mathcal{B}'}$ are linearly independent, showing that $\mathcal{B}'$ is a basis.

We now show that $\mathcal{C}(\phi, \mathcal{B})$ is adjacent to $\mathcal{C}(\phi, \mathcal{B}')$ along the facet *cone* $\left(G(\phi)^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)$. Notice that for any set $J \subset \mathcal{E}$ and $\sigma \in S_{\sigma}$, $q(\phi, \sigma)$ lies in the relative interior of *cone* $(G(\phi)_{\bullet J})$ if and only if $q(\phi, \sigma)$ is a strictly positive combination of the columns of $G(\phi)_{\bullet J}$, i.e., for each $\gamma \in J$ there exists $\beta_{\gamma} > 0$ such that $q(\phi, \sigma) = \sum\limits_{\gamma \in J} \beta_{\gamma} G(\phi)_{\bullet\gamma}$. Thus, consider

$$\begin{aligned} \mathcal{W}(\beta) :&= \sum_{\gamma\in(\mathcal{B}\setminus\{i\})} \beta_{\gamma} G(\phi)_{\bullet\gamma} \\ &= \beta_j G(\phi)_{\bullet j} + \sum_{\gamma\in(\mathcal{B}\setminus\{i,j\})} \beta_{\gamma} G(\phi)_{\bullet\gamma} \end{aligned} \tag{5.79}$$

Recall that $(T_{\mathcal{B}}(\phi))_{\bullet\bar{\imath}} = G(\phi)^{-1}_{\bullet\mathcal{B}} G(\phi)_{\bullet\bar{\imath}}$, which implies:

$$\begin{aligned} G(\phi)_{\bullet\bar{\imath}} &= G(\phi)_{\bullet\mathcal{B}}(T_{\mathcal{B}}(\phi))_{\bullet\bar{\imath}} \\ &= \sum_{\gamma\in\mathcal{B}} G(\phi)_{\bullet\gamma}(T_{\mathcal{B}}(\phi))_{\gamma\bar{\imath}} \\ &= \sum_{\gamma\in(\mathcal{B}\setminus\{i\})} G(\phi)_{\bullet\gamma}(T_{\mathcal{B}}(\phi))_{\gamma\bar{\imath}} \\ &= G(\phi)_{\bullet j}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} + \sum_{\gamma\in(\mathcal{B}\setminus\{i,j\})} G(\phi)_{\bullet\gamma}(T_{\mathcal{B}}(\phi))_{\gamma\bar{\imath}} \end{aligned} \tag{5.80}$$

Since $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$, (5.80) gives $G(\phi)_{\bullet j} = \dfrac{1}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}} G(\phi)_{\bullet\bar{\imath}} - \sum\limits_{\gamma\in(\mathcal{B}\setminus\{i,j\})} G(\phi)_{\bullet\gamma} \dfrac{(T_{\mathcal{B}}(\phi))_{\gamma\bar{\imath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}$.

Using this result and substituting into (5.79) yields

$$\begin{aligned}\mathcal{W}(\beta) &= \beta_j \left(\frac{1}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}} G(\phi)_{\bullet \bar{\imath}} - \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} G(\phi)_{\bullet \gamma} \frac{(T_{\mathcal{B}}(\phi))_{\gamma \bar{\imath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}} \right) + \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} \beta_\gamma G(\phi)_{\bullet \gamma} \\ &= \frac{\beta_j}{(T_{\mathcal{B}})_{j,\bar{\imath}}} G(\phi)_{\bullet \bar{\imath}} - \sum_{\gamma \in (\mathcal{B} \setminus \{i,j\})} \left(\beta_\gamma - \beta_j \frac{(T_{\mathcal{B}}(\phi))_{\gamma \bar{\imath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}} \right) G(\phi)_{\bullet \gamma}\end{aligned} \tag{5.81}$$

From (5.79) and (5.81) we observe that by selecting $\tilde{\beta}$ so that: (i) $\tilde{\beta}_\gamma > 0$ for all $\gamma \in (\mathcal{B} \setminus \{i\})$, and (ii) $\tilde{\beta}_\gamma > \tilde{\beta}_j \dfrac{(T_{\mathcal{B}}(\phi))_{\gamma \bar{\imath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}$ for all $\gamma \in (\mathcal{B} \setminus \{i,j\})$, we have that $\mathcal{W}(\tilde{\beta})$ is in the relative interior of both *cone* $\left(G(\phi)_{\bullet\,(\mathcal{B} \setminus \{i\})}\right)$ and *cone* $\left(G(\phi)_{\bullet\,(\mathcal{B}' \setminus \{\bar{\jmath}\})}\right)$. This shows that $dim(cone(G(\phi)_{\bullet\,(\mathcal{B} \setminus \{i\})}) \cap cone(G(\phi)_{\bullet\,(\mathcal{B}' \setminus \{\bar{\jmath}\})})) = h - 1$ and therefore $\mathcal{B}$ and $\mathcal{B}'$ are adjacent.

($\Rightarrow$): We prove this direction by contradiction. Consider the following 3 cases:

Case 1: $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} \neq 0$

By Corollary 5.59, $\hat{\mathcal{B}} = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is the unique basis such that $\mathcal{C}(\phi, \hat{\mathcal{B}})$ is adjacent to $\mathcal{C}(\phi, \mathcal{B})$ along the facet *cone* $\left(G(\phi)^{-1}_{\bullet(\mathcal{B} \setminus \{i\})}\right)$. This is a contradiction.

Case 2: $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} \leq 0$

Since $\mathcal{B}'$ is a basis, the unique way to represent $\mathcal{W}(\beta)$ as a linear combination of the columns of $G(\phi)_{\bullet \mathcal{B}'}$ is (5.81). Therefore, in this case there does not exist β such that $\mathcal{W}(\beta)$ lies in both the relative interiors of *cone* $\left(G(\phi)^{-1}_{\bullet(\mathcal{B} \setminus \{i\})}\right)$ and $\mathcal{C}(\phi, \mathcal{B}')$. Hence, $\mathcal{C}(\phi, \mathcal{B})$ cannot be adjacent to $\mathcal{C}(\phi, \mathcal{B}')$ along *cone* $\left(G(\phi)^{-1}_{\bullet(\mathcal{B} \setminus \{i\})}\right)$, which is a contradiction.

Case 3: $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} = 0$

Since $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} = 0$, the matrix $G(\phi)^{-1}_{\bullet \mathcal{B}} G(\phi)_{\bullet \mathcal{B}'}$ has a row of all zeros. Thus $G(\phi)^{-1}_{\bullet \mathcal{B}} G(\phi)_{\bullet \mathcal{B}'}$ is not invertible, which is a contradiction since both $\mathcal{B}$ and $\mathcal{B}'$ are bases.

Finding a contradiction in each of the cases above shows that we must have $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} = 0$, $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$, and $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} \neq 0$. □

By combining the results of Propositions 5.54, 5.58, 5.60 and 5.61 as well as Observation 5.57 and Corollary 5.59, we are now able to develop the following strategy for finding all complementary bases which are adjacent to a given basis $\mathcal{B}$:

1. Calculate the tableau $T_{\mathcal{B}}(\phi)$ associated with basis $\mathcal{B}$.
2. For any $i \in \mathcal{B}$ for which there exists $\phi \in S_\phi$ such that $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} \neq 0$, the set $(\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a complementary basis adjacent to $\mathcal{B}$.

3. For any distinct $i, j \in \mathcal{B}$ for which there exists $\phi \in \mathcal{IR}^{\phi}_{\mathcal{B}}$ such that $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}} = 0$, $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$ and $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} \neq 0$, the set $(\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a feasible complementary basis adjacent to $\mathcal{B}$.

Recall, however, that our goal is to partition S_θ. Although there is a strong relationship between complementary cones and invariancy regions, it is not always the case that adjacent invariancy regions result from adjacent complementary cones. Recall, though, that in Proposition 5.53 we showed that for any pair of adjacent full dimensional invariancy regions $\mathcal{IR}_{\mathcal{B}_i}$ and $\mathcal{IR}_{\mathcal{B}_j}$ with $i < j$, there always exists a sequence of bases $\{\mathcal{B}_n\}_{n=i+1}^{j-1}$ and $\Phi \subseteq \cap_{n=i}^{j} \mathcal{IR}^{\phi}_{\mathcal{B}_n}$ such that $\mathcal{C}(\phi, \mathcal{B}_\gamma)$ and $\mathcal{C}(\phi, \mathcal{B}_{\gamma+1})$ are adjacent for all $\gamma \in \{i, \ldots, j-1\}$ and $\phi \in \Phi$, and $dim\left(\mathcal{IR}_{\mathcal{B}_\gamma}\right) = k - 1$ for all $\gamma \in \{i+1, \ldots, j-1\}$. Throughout the rest of this work we utilize the theory developed thus far in order to establish a method for determining such sequences. We first present a brief outline of this method and then provide the details necessary for its implementation.

Outline 5.62:

Preliminaries: Let $\mathcal{R}$ represent a set of invariancy regions for which adjacent regions need to be found. Let $\mathcal{K}$ represent the set of feasible bases discovered.

Phase 1 – Initialization: Find an initial full dimensional invariancy region $\mathcal{IR}_{\mathcal{B}_0}$ and add it to $\mathcal{R}$. Add $\mathcal{B}_0$ to $\mathcal{K}$.

Phase 2 – Main Step:

1. Select $\mathcal{IR}_{\mathcal{B}} \in \mathcal{R}$ and remove it from $\mathcal{R}$.
2. Determine the set of feasible bases which have invariancy regions that are: (i) adjacent to $\mathcal{IR}_{\mathcal{B}}$, and (ii) at least $(k-1)$–dimensional.
3. For each basis $\hat{\mathcal{B}}$ discovered in step 2, if $\hat{\mathcal{B}} \notin \mathcal{K}$, add $\mathcal{IR}_{\hat{\mathcal{B}}}$ to $\mathcal{R}$ and $\hat{\mathcal{B}}$ to $\mathcal{K}$.
4. If $\mathcal{R} = \emptyset$, STOP. Otherwise, go back to step 1.

Note that in Outline 5.62 the main step serves as a basis for Algorithm 5.5, while the initialization phase serves as a basis for Algorithm 5.12. The former is presented in Section 5.4 and the latter in Section 5.5. We now discuss the details necessary to implement the procedure above. The details of the initialization step, i.e., determining an initial feasible basis $\mathcal{B}_0$ such that

$dim\,(\mathcal{IR}_{\mathcal{B}_0}) \geq k-1$, are given in Section 5.5. For any feasible complementary basis $\mathcal{B}$ and index $i \in \mathcal{B}$, we define the following sets which will be useful during the remainder of this discussion:

$$\mathrm{Z}_{\mathcal{B}} := \Big\{ j \in \mathcal{B} : \left(Adj\,(G(\phi)_{\bullet\mathcal{B}})\right)_{j\,\bullet} q(\phi,\sigma) = 0 \quad \forall\, \phi \in S_\theta, \sigma \in S_\sigma \Big\} \tag{5.82}$$

$$\mathscr{h}^{i}_{\mathcal{B}} := \left\{ (\phi,\sigma) \in \mathbb{R}^k : \left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{i\,\bullet} q(\phi,\sigma) = 0 \right\} \tag{5.83}$$

$$\mathrm{E}_{\mathcal{B}} := \Big\{ j \in \mathcal{B} : \mathscr{h}^{j}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}} = \emptyset \Big\} \tag{5.84}$$

$$\mathrm{H}^{i}_{\mathcal{B}} := \Big\{ j \in \mathcal{B} \setminus (\mathrm{E}_{\mathcal{B}} \cup \{i\}) : \Big(\mathscr{h}^{i}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\Big) \subseteq \Big(\mathscr{h}^{j}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\Big) \Big\} \tag{5.85}$$

Here $\mathrm{Z}_{\mathcal{B}}$ is the set of indices in $\mathcal{B}$ for which the RHS of $T_{\mathcal{B}}(\phi,\sigma)$ is identically zero. For given $\phi \in S_\theta$ and $\sigma \in S_\sigma$ these RHS values can be interpreted as the multipliers on the columns of $G(\phi)_{\bullet\,\mathcal{B}}$ needed to represent $q(\phi,\sigma)$ as a linear combination of the columns of $G(\phi)$. Thus, if the RHS value is identically zero for some index i, this indicates that the column $G(\phi)_{\bullet\,i}$ is unnecessary in the representation of $q(\phi,\sigma)$. There is also another interpretation. Notice from (5.8) that for each $i \in \mathcal{B}$, $\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\,\bullet} q(\phi,\sigma) \geq 0$ is a defining inequality of $\mathcal{IR}_{\mathcal{B}}$. Thus if there is some $i \in \mathcal{B}$ for which the RHS of $T_{\mathcal{B}}(\phi,\sigma)$ is identically zero, the associated defining inequality of $\mathcal{IR}_{\mathcal{B}}$ is $0 \geq 0$, which is trivially satisfied. Now consider $\mathscr{h}^{i}_{\mathcal{B}}$. Given an index $i \in \mathcal{B}$, $\mathscr{h}^{i}_{\mathcal{B}}$ is the hypersurface in $\mathbb{R}^k$ which implies the defining constraint of $\mathcal{IR}_{\mathcal{B}}$ associated with index i. The set $\mathrm{E}_{\mathcal{B}}$ is then the set of indices $j \in \mathcal{B}$ for which $\mathscr{h}^{j}_{\mathcal{B}}$ does not intersect $\mathcal{IR}_{\mathcal{B}}$. Finally, consider $\mathrm{H}^{i}_{\mathcal{B}}$. Given an index $i \in \mathcal{B}$, the set $\mathrm{H}^{i}_{\mathcal{B}}$ is the set of indices in $\mathcal{B} \setminus (\mathrm{E}_{\mathcal{B}} \cup \{i\})$ such that the intersection of $\mathscr{h}^{i}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}}$ is a subset of the intersection of $\mathscr{h}^{j}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}}$. Recognize that given $i \in \mathcal{B}$, for each $j \in \mathrm{H}^{i}_{\mathcal{B}} \subseteq \mathcal{B} \setminus (\mathrm{E}_{\mathcal{B}} \cup \{i\})$, every point in $\mathcal{IR}_{\mathcal{B}}$ which satisfies the defining constraint of $\mathcal{IR}_{\mathcal{B}}$ associated with i at equality also satisfies the defining constraints of $\mathcal{IR}_{\mathcal{B}}$ associated with j at equality. Now, given a complementary basis $\mathcal{B}$, consider the construction of the sets $\mathrm{Z}_{\mathcal{B}}$, $\mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^{i}_{\mathcal{B}}$ for each $i \in \mathcal{B}$. Recognize from (5.82) that $\mathrm{Z}_{\mathcal{B}}$ can be constructed easily by observing $T_{\mathcal{B}}(\phi,\sigma)$. Unfortunately, constructing $\mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^{i}_{\mathcal{B}}$ (for a given $i \in \mathcal{B}$) is not so straightforward. For this purpose we introduce the following two propositions.

Proposition 5.63. Given a complementary basis $\mathcal{B}$ for which $dim(\mathcal{IR}_{\mathcal{B}}) \geq k-1$ and distinct indices $i, j \in \mathcal{B}$, $j \in \mathrm{H}^i_{\mathcal{B}}$ if and only if the following nonlinear program has an optimal value of zero.

$$\begin{array}{rll} NLP_H(\mathcal{B}, i, j) := & \max\limits_{\lambda,\phi,\sigma} & \lambda \\ & \text{s.t.} & g(\mathcal{B})\,(Adj(G(\phi)_{\bullet\mathcal{B}}))_{\ell\bullet}\,q(\phi,\sigma) \geq 0 \quad \forall\, \ell \in (\mathcal{B} \setminus (\mathrm{Z}_{\mathcal{B}} \cup \{i, j\})) \\ & & (Adj(G(\phi)_{\bullet\mathcal{B}}))_{i\bullet}\,q(\phi,\sigma) = 0 \\ & & g(\mathcal{B})\,(Adj(G(\phi)_{\bullet\mathcal{B}}))_{j\bullet}\,q(\phi,\sigma) \geq \lambda \\ & & \phi \in S_\phi, \;\; \sigma \in S_\sigma \end{array} \tag{5.86}$$

Proof. ($\Rightarrow$): Let $(\lambda^*, \phi^*, \sigma^*)$ be an optimal solution to $NLP_H(\mathcal{B}, i, j)$. Notice that since $j \in \mathrm{H}^i_{\mathcal{B}}$, every $\theta \in (\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}})$ is also in $(\hbar^j_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}})$. This means that for every (λ, ϕ, σ) feasible to $NLP_H(\mathcal{B}, i, j)$, $(Adj(G(\phi)_{\bullet\mathcal{B}}))_{j\bullet}\,q(\phi,\sigma) = 0$, which shows that $\lambda^* \leq 0$. Additionally, since $\dim(\mathcal{IR}_{\mathcal{B}}) \geq k-1$, there must exist $\theta' = (\phi', \sigma')$ in $\mathcal{IR}_{\mathcal{B}}$, i.e., all defining inequalities of $\mathcal{IR}_{\mathcal{B}}$ are satisfied at θ'. Thus $(0, \phi', \sigma')$ is feasible to $NLP_H(\mathcal{B}, i, j)$. This shows that $\lambda^* \geq 0$. Thus, we must have $\lambda^* = 0$.

($\Leftarrow$): Suppose $(\lambda^*, \phi^*, \sigma^*)$ is an optimal solution to $NLP_H(\mathcal{B}, i, j)$ and $\lambda^* = 0$. Recognize that if there existed $\theta' = (\phi', \sigma')$ in $\mathcal{IR}_{\mathcal{B}}$ such that $\theta' \in \hbar^i_{\mathcal{B}}$ but $\theta' \notin \hbar^j_{\mathcal{B}}$, then there would also exist $\lambda' > 0$ such that $g(\mathcal{B})\,(Adj(G(\phi)_{\bullet\mathcal{B}}))_{j\bullet}\,q(\phi,\sigma) = \lambda'$. Furthermore, $(\lambda', \phi', \sigma')$ would be feasible for $NLP_H(\mathcal{B}, i, j)$. This contradicts the fact that $\lambda^* = 0$, though, and so we must have that for all $\theta \in (\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}})$, $\theta \in \hbar^j_{\mathcal{B}}$. Hence, $\left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right) \subseteq \left(\hbar^j_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right)$ and therefore $j \in \mathrm{H}^i_{\mathcal{B}}$. □

We note here that due to the fact that S_θ is assumed to be a bounded polytope, NLP_H will always have a bounded feasible region. Moreover, S_θ being bounded guarantees that every NLP we introduce in this section will also have a bounded feasible region.

Proposition 5.64. Given a complementary basis $\mathcal{B}$ for which $dim(\mathcal{IR}_{\mathcal{B}}) \geq k-1$ and an index $i \in \mathcal{B}$, we have $i \in \mathrm{E}_{\mathcal{B}}$ if and only if there exists $j \in \mathcal{B}$ such that $NLP_H(\mathcal{B}, i, j)$ has a strictly negative optimal value, or is infeasible.

Proof. ($\Rightarrow$): Recall from (5.84) that since $i \in \mathrm{E}_{\mathcal{B}}$, we have $\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}} = \emptyset$. This shows that there does not exist a $j \in \mathcal{B}$ with a feasible solution $(\lambda^j, \phi^j, \sigma^j)$ to $NLP_H(\mathcal{B}, i, j)$ for which $\lambda^j \geq 0$. Now, for each $j \in \mathcal{B}$ let $\overline{\mathcal{IR}}^j_{\mathcal{B}}$ represent the semi-algebraic superset of $\mathcal{IR}_{\mathcal{B}}$ which results from eliminating the defining constraint of $\mathcal{IR}_{\mathcal{B}}$ associated with index j from $\mathcal{IR}_{\mathcal{B}}$. For an arbitrary $j \in \mathcal{B}$, suppose that $\hbar^i_{\mathcal{B}} \cap \overline{\mathcal{IR}}^j_{\mathcal{B}} = \emptyset$. Clearly, in this case $NLP_H(\mathcal{B}, i, j)$ is infeasible. On the other hand, if $\hbar^i_{\mathcal{B}} \cap \overline{\mathcal{IR}}^j_{\mathcal{B}} \neq \emptyset$, let $\tilde{\theta}^j = (\tilde{\phi}^j, \tilde{\sigma}^j)$ be a point in $\hbar^i_{\mathcal{B}} \cap \overline{\mathcal{IR}}^j_{\mathcal{B}}$. Then we have: (i) $g(\mathcal{B})\left(Adj(G(\tilde{\phi}^j)_{\bullet\mathcal{B}})\right)_{\ell\bullet} q(\tilde{\phi}^j, \tilde{\sigma}^j) \geq 0$

for all $\ell \in (\mathcal{B} \setminus \{i, j\})$, (ii) $\left(Adj(G(\tilde{\phi}^j)_{\bullet\mathcal{B}})\right)_{i\bullet} q(\tilde{\phi}^j, \tilde{\sigma}^j) = 0$, and (iii) there exists $\tilde{\lambda}^j < 0$ such that $g(\mathcal{B}) \left(Adj(G(\tilde{\phi}^j)_{\bullet\mathcal{B}})\right)_{j\bullet} q(\tilde{\phi}^j, \tilde{\sigma}^j) \geq \tilde{\lambda}^j$. Thus, $NLP_H(\mathcal{B}, i, j)$ is feasible and has a strictly negative optimal value. We have now shown that for each $j \in \mathcal{B}$: (i) if $\hbar^i_{\mathcal{B}} \cap \overline{\mathcal{IR}}^j_{\mathcal{B}} = \emptyset$ then $NLP_H(\mathcal{B}, i, j)$ is infeasible, and (ii) if $\hbar^i_{\mathcal{B}} \cap \overline{\mathcal{IR}}^j_{\mathcal{B}} \neq \emptyset$ then $NLP_H(\mathcal{B}, i, j)$ is feasible, but has a strictly negative optimal value.

($\Leftarrow$): Recognize that if there existed $\theta' = (\phi', \sigma')$ in $\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}$ then there would also exist $\lambda' \geq 0$ such that $g(\mathcal{B}) \left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{\ell\bullet} q(\phi, \sigma) = \lambda'$ for all $\ell \in \mathcal{B}$. Furthermore, $(\lambda', \phi', \sigma')$ would be feasible to $NLP_H(\mathcal{B}, i, \ell)$ for all $\ell \in \mathcal{B}$. However, this contradicts the fact that there exists $j \in \mathcal{B}$ for which the optimal value of $NLP_H(\mathcal{B}, i, j)$ is strictly negative. Hence, we must have that $(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}) = \emptyset$, which shows that $i \in \mathrm{E}_{\mathcal{B}}$. □

The results of Propositions 5.63 and 5.64 provide a strategy so that, given a complementary basis $\mathcal{B}$, we can build the sets $\mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^i_{\mathcal{B}}$ for each $i \in \mathcal{B}$. We present this strategy in Algorithm 5.1.

Algorithm 5.1 BUILDEANDH($\mathcal{B}$) – Build $\mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^i_{\mathcal{B}}$ for each $i \in \mathcal{B}$.
Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}_{\mathcal{B}}) \geq k - 1$. (Assume the set $\mathrm{Z}_{\mathcal{B}}$ has been constructed.)
Output: The sets $\mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^i_{\mathcal{B}}$ for each $i \in \mathcal{B}$.

```
1: Let $\mathrm{E}_{\mathcal{B}} = \emptyset$.
2: Let $\mathrm{H}^{\ell}_{\mathcal{B}} = \emptyset$ for each $\ell \in \mathcal{B}$.
3: for $i \in (\mathcal{B} \setminus (\mathrm{Z}_{\mathcal{B}} \cup \mathrm{E}_{\mathcal{B}}))$ do
4:     for $j \in (\mathcal{B} \setminus (\mathrm{Z}_{\mathcal{B}} \cup \mathrm{E}_{\mathcal{B}} \cup \{i\}))$ do
5:         if $j \notin \mathrm{H}^i_{\mathcal{B}}$ then solve $NLP_H(\mathcal{B}, i, j)$ to obtain optimal solution $(\lambda^*, \phi^*, \sigma^*)$.
6:             if $\lambda^* = 0$ then add $\left(j \cup \mathrm{H}^j_{\mathcal{B}}\right)$ to $\mathrm{H}^i_{\mathcal{B}}$.
7:             else if $\lambda^* < 0$ then add $i$ to $\mathrm{E}_{\mathcal{B}}$ and exit the for loop beginning on Line 4.
8: Return $\mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^{\ell}_{\mathcal{B}}$ for each $\ell \in \mathcal{B}$.
```

Throughout the remainder of this section we introduce several more propositions whose results allow us to perform the steps of Outline 5.62. Similarly to Proposition 5.63, many of these propositions introduce nonlinear programs (NLPs) that can be solved in order to determine, for example, if a given invariancy region is full dimensional. We note that many of these NLPs do not need to be solved to optimality, but rather a feasible solution must be found which has an associated objective function value which is strictly positive. As we develop the theory necessary for partitioning S_θ, we also show directly how this theory can be applied to each of the instances we introduced earlier in Examples 5.5 and 5.6. We will do so, though, using the subvectors ϕ and σ rather than θ. Consider (5.5) and (5.6) and recall that ϕ represents the subvector of θ such that every element of ϕ is present in some element of $M(\theta)$ and σ represents the subvector of θ such that no element of σ is present in any element of $M(\theta)$. Hence, we have: (i) for Example 5.5 , $\phi = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ and $\sigma = \emptyset$, and (ii) for Example 5.6, $\phi = \theta_1$ and $\sigma = \theta_2$. For the sake of clarity, when discussing these examples we use variable names to describe the elements of each basis rather than their corresponding indices. The solution to Example 5.5 is given in Table 5.1 and depicted in Figure 5.2a while the solution to Example 5.6 is given in Table 5.2 and depicted in Figure 5.2b. How each solution is determined is shown during the discussion that follows. We note that each nonlinear program solved throughout the course of this work is solved using the "fmincon" function in MATLAB.

Due to the size of the tableaux we utilize throughout this section and Section 5.5, we do not include them in their entirety in these sections. Instead, we include tableaux for Examples 5.5 and 5.6 in Sections 5.A and 5.B, respectively. We note that for Example 5.6, phase 1 of our procedure requires only one iteration and thus, all tables in Section 5.B are dedicated to phase 2. On the other hand, for Example 5.5, phase 1 requires several iterations and for this reason Tables 5.4–5.10 are dedicated to phase 1, while Tables 5.11–5.13 are dedicated to phase 2. Additionally, the size of several of the tableaux for Example 5.5 are so large that we cannot include every column. In these cases (Tables 5.8–5.13) we include only the columns associated with nonbasic variables (note that the columns for basic variables are simply identity vectors). We also point out that each of the tables in Sections 5.A and 5.B is not necessary obtained from the previous table. For example, bases $\mathcal{B}_i^{5.5}$, $\mathcal{B}_{ii}^{5.5}$, and $\mathcal{B}_{iii}^{5.5}$ are all obtained from exchange pivots from $\mathcal{B}^{*5.5}$ and thus Tables 5.5–5.7 are all obtained from Table 5.4. As another example, we note that $\mathcal{B}_2^{5.5}$ is obtained from a diagonal pivot from basis $\mathcal{B}_0^{5.5}$ and so Table 5.13 is obtained from Table 5.11. How each table is obtained from previous tables will be made clear as we consider Examples 5.5 and 5.6 in more detail in this section and Section 5.5.

For Examples 5.5 and 5.6 we claim that the initial bases $\mathcal{B}_0^{5.5} = \{w_1, z_2, z_3, z_4, z_5\}$ and $\mathcal{B}_0^{5.6} = \{w_1, w_2, w_3, w_4\}$ are feasible. Respective tableaux for $\mathcal{B}_0^{5.5}$ and $\mathcal{B}_0^{5.6}$ are contained in Tables 5.11 and 5.14 which are found in Sections 5.A and 5.B, respectively. A discussion on obtaining initial bases is provided in Section 5.5. The following proposition provides the tools necessary for determining whether or not the invariancy region associated with a given basis is full dimensional.

Table 5.1: Solution for Example 5.5.

$$\mathcal{B}_0^{5.5}: \quad \left[\begin{array}{l} w_1 = \frac{3\,\phi_1^3+18\,\phi_1^2\,\phi_2-49\,\phi_1^2-75\,\phi_1\,\phi_2^2+148\,\phi_1\,\phi_2+68\,\phi_1+96\,\phi_2^2-16\,\phi_2-76}{2\left(-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22\right)} \\ z_2 = -\frac{(\phi_1+2)\left(9\,\phi_1^3-9\,\phi_1^2\,\phi_2-33\,\phi_1^2-87\,\phi_1\,\phi_2^2+21\,\phi_1\,\phi_2+22\,\phi_1-59\,\phi_2^3+13\,\phi_2^2+50\,\phi_2+5\right)}{4\left(-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22\right)} \\ z_3 = \frac{(\phi_1+2)\left(-6\,\phi_1^2-\phi_1\,\phi_2+11\,\phi_1+15\,\phi_2^2-16\,\phi_2+1\right)}{2\left(-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22\right)} \\ z_4 = \frac{(\phi_1+2)\left(3\,\phi_1^2+8\,\phi_1\,\phi_2-\phi_1+5\,\phi_2^2+5\,\phi_2-11\right)}{2\left(-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22\right)} \\ z_5 = \frac{(\phi_1+2)\left(9\,\phi_1-13\,\phi_2+\phi_1\,\phi_2+21\,\phi_2^2-12\right)}{2\left(-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22\right)} \end{array}\right]$$

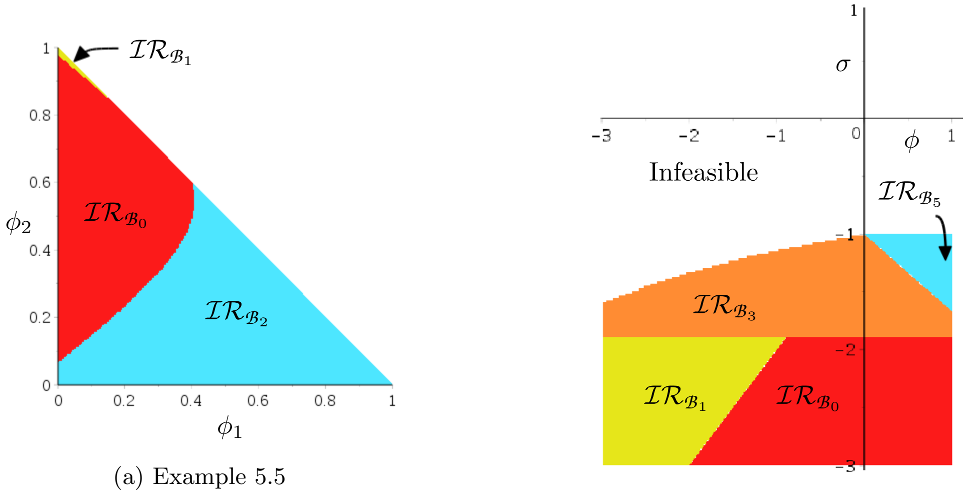
$$\mathcal{B}_1^{5.5}: \quad \left[\begin{array}{l} z_1 = \frac{3\,\phi_1^3+18\,\phi_1^2\,\phi_2-49\,\phi_1^2-75\,\phi_1\,\phi_2^2+148\,\phi_1\,\phi_2+68\,\phi_1+96\,\phi_2^2-16\,\phi_2-76}{8\,(32\,\phi_2-7\,\phi_1+29)} \\ z_2 = \frac{-81\,\phi_1^3+288\,\phi_1^2\,\phi_2-111\,\phi_1^2+47\,\phi_1\,\phi_2^2+856\,\phi_1\,\phi_2+422\,\phi_1+544\,\phi_2^2+392\,\phi_2-44}{16\,(32\,\phi_2-7\,\phi_1+29)} \\ z_3 = -\frac{61\,\phi_1-60\,\phi_1\,\phi_2+29\,\phi_1^2-12}{4\,(32\,\phi_2-7\,\phi_1+29)} \\ z_4 = \frac{75\,\phi_1+32\,\phi_2-5\,\phi_1\,\phi_2+13\,\phi_1^2+20}{4\,(32\,\phi_2-7\,\phi_1+29)} \\ z_5 = \frac{16\,\phi_1+96\,\phi_2+9\,\phi_1\,\phi_2+2\,\phi_1^2+84}{4\,(32\,\phi_2-7\,\phi_1+29)} \end{array}\right]$$

$$\mathcal{B}_2^{5.5}: \quad \left[\begin{array}{l} w_1 = \frac{-11\,\phi_1^2+37\,\phi_1+48\,\phi_2-44}{2\,(7\,\phi_1+8\,\phi_2-13)} \\ z_2 = -\frac{(\phi_1+2)\left(-9\,\phi_1^2-23\,\phi_1\,\phi_2+15\,\phi_1-17\,\phi_2^2+23\,\phi_2+3\right)}{4\,(7\,\phi_1+8\,\phi_2-13)} \\ w_3 = -\frac{(\phi_1+2)\left(-6\,\phi_1^2-\phi_1\,\phi_2+11\,\phi_1+15\,\phi_2^2-16\,\phi_2+1\right)}{2\,(7\,\phi_1+8\,\phi_2-13)} \\ z_4 = \frac{(\phi_1+2)\,(3\,\phi_1+5\,\phi_2-6)}{2\,(7\,\phi_1+8\,\phi_2-13)} \\ z_5 = \frac{(\phi_1+2)\,(4\,\phi_1+3\,\phi_2-7)}{2\,(7\,\phi_1+8\,\phi_2-13)} \end{array}\right]$$

Table 5.2: Solution for Example 5.6.

$$\mathcal{B}_0^{5.6}: \quad \left[\begin{array}{l} w_1 = -\sigma - 1 \\ w_2 = \phi - \sigma - 1 \\ w_3 = -18\,\sigma - 34 \\ w_4 = -9\,\sigma - 17 \end{array}\right] \qquad \mathcal{B}_1^{5.6}: \quad \left[\begin{array}{l} w_1 = -\frac{8\,\sigma+\phi\,\sigma+8}{\phi+7} \\ z_2 = -\frac{9\,\sigma+17}{\phi+5} \\ w_3 = -\frac{(2\,\phi+13)\,(9\,\sigma+17)}{\phi+5} \\ z_4 = \frac{\sigma-\phi+1}{\phi+7} \end{array}\right]$$

$$\mathcal{B}_3^{5.6}: \quad \left[\begin{array}{l} z_1 = \frac{(2\,\phi+13)\,(9\,\sigma+17)}{\phi+8} \\ z_2 = \frac{9\,\sigma+17}{\phi+8} \\ z_3 = -\frac{8\,\sigma+\phi\,\sigma+8}{2\,\phi+19} \\ z_4 = -\frac{\phi+\frac{3}{2}\sigma+\frac{3}{2}}{\phi+\frac{19}{2}} \end{array}\right] \qquad \mathcal{B}_5^{5.6}: \quad \left[\begin{array}{l} z_1 = 18\,\sigma + 34 \\ w_2 = \phi + \frac{3}{2}\sigma + \frac{3}{2} \\ z_3 = -\frac{1}{2}\sigma - \frac{1}{2} \\ w_4 = 9\,\sigma + 17 \end{array}\right]$$



(a) Example 5.5

(b) Example 5.6

Figure 5.2: Partitions of S_θ for the two examples.

Proposition 5.65. For a given feasible complementary basis $\mathcal{B}$, $\mathcal{IR}_{\mathcal{B}}$ is full dimensional if and only if $|\mathrm{Z}_{\mathcal{B}}| \le h-(k-p)$ and there exists $\phi' \in S_\phi$ and $\sigma' \in S_\sigma$ such that $g(\mathcal{B})\left(Adj(G(\phi')_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi',\sigma') > 0$ for all $i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}}$, i.e., if the following NLP has a strictly positive optimal value:

$$NLP_D(\mathcal{B}) := \begin{array}{lll} \max\limits_{\lambda,\phi,\sigma} & \lambda & \\ \text{s.t.} & g(\mathcal{B})\left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi,\sigma) \ge \lambda\mathbf{1} & \forall\, i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}} \\ & \phi \in S_\phi,\ \ \sigma \in S_\sigma & \end{array} \tag{5.87}$$

Proof. ($\Leftarrow$): Suppose $(\lambda^*,\phi^*,\sigma^*)$ is a solution to $NLP_D(\mathcal{B})$ and $\lambda^* > 0$. Then since $g(\mathcal{B})\left(Adj(G(\phi')_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi',\sigma') > 0$ for all $i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}}$, we have that $\mathcal{AS}(\phi^*)$ intersects the relative interior of $cone\left(G(\phi^*)_{\bullet(\mathcal{B}\setminus \mathrm{Z}_{\mathcal{B}})}\right)$. Furthermore, since $\mathcal{AS}(\phi^*)$ intersects the relative interior of $cone\left(G(\phi^*)_{\bullet(\mathcal{B}\setminus \mathrm{Z}_{\mathcal{B}})}\right)$, there must exist $\epsilon > 0$ such that $\mathcal{AS}(\phi)$ intersects the relative interior of $cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus \mathrm{Z}_{\mathcal{B}})}\right)$ for all $\phi \in B_\epsilon(\phi^*)$. Thus, since $dim(\mathcal{AS}(\phi)) = k-p$ for all $\phi \in S_\phi$, we have $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi,\mathcal{B})\right) = \min\left\{k-p, h-|\mathrm{Z}_{\mathcal{B}}|\right\}$ for all $\phi \in B_\epsilon(\phi^*)$. By Corollary 5.51 this shows that $dim(\mathcal{IR}_{\mathcal{B}}) = k$ since $|\mathrm{Z}_{\mathcal{B}}| \le h-(k-p)$.

($\Rightarrow$): We prove this direction using contradiction. Consider the following cases:

Case 1: $|\mathrm{Z}_{\mathcal{B}}| > h-(k-p)$

Notice that for each $i \in \mathrm{Z}_{\mathcal{B}}$, the column $G(\phi)_{\bullet i}$ is unnecessary for the description of $\mathcal{AS}(\phi) \cap \mathcal{C}(\phi,\mathcal{B})$ for all $\phi \in S_\phi$. Therefore $\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi,\mathcal{B})\right) \subseteq cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus \mathrm{Z}_{\mathcal{B}})}\right)$ for all $\phi \in S_\phi$. However, since $|\mathrm{Z}_{\mathcal{B}}| > h-(k-p)$, $dim\left(cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus \mathrm{Z}_{\mathcal{B}})}\right)\right) < k-p$ for all $\phi \in S_\phi$. This shows

that $dim\,(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})) < k - p$ for all $\phi \in S_\phi$ and thus, by Corollary 5.51, $\mathcal{IR}_{\mathcal{B}}$ cannot be full dimensional, which is a contradiction.

<u>Case 2</u>: There does not exist a solution $(\lambda^*, \phi^*, \sigma^*)$ to $NLP_D(\mathcal{B})$ such that $\lambda^* > 0$.

If $NLP_D(\mathcal{B})$ is infeasible, $\mathcal{IR}_{\mathcal{B}}$ is empty, which is a contradiction. On the other hand, suppose that for all solutions $(\lambda^*, \phi^*, \sigma^*)$ to $NLP_D(\mathcal{B})$, $\lambda^* \leq 0$. If this is the case then for all $\theta \in \mathcal{IR}_{\mathcal{B}}$ and all $\epsilon > 0$, we have $B_\epsilon(\theta) \not\subset \mathcal{IR}_{\mathcal{B}}$. Therefore $\mathcal{IR}_{\mathcal{B}}$ cannot be full dimensional, which is a contradiction.

Finding contradictions in both cases above shows that there must exist a solution $(\lambda^*, \phi^*, \sigma^*)$ to $NLP_D(\mathcal{B})$ such that $\lambda^* > 0$ and that $|\mathrm{Z}_{\mathcal{B}}| \leq h - (k - p)$. □

As a result of Proposition 5.65, for any basis $\mathcal{B}$, $NLP_D(\mathcal{B})$ can be used to determine whether or not $\mathcal{IR}_{\mathcal{B}}$ is full dimensional. $NLP_D(\mathcal{B}_0^{5.5})$ and $NLP_D(\mathcal{B}_0^{5.6})$ can be derived from (5.12) and (5.13). Note that obtaining these bases as well as the representations of their respective invariancy regions will be discussed in Section 5.5.

$NLP_D(\mathcal{B}_0^{5.5})$:

$$\begin{array}{lc}
\max\limits_{\lambda,\phi} & \lambda \\
\text{s.t.} & -3\phi_1^3 - 18\phi_1^2\phi_2 + 49\phi_1^2 + 75\phi_1\phi_2^2 - 148\phi_1\phi_2 - 68\phi_1 - 96\phi_2^2 + 16\phi_2 + 76 \geq \lambda \\
& (\phi_1 + 2)(9\phi_1^3 - 9\phi_1^2\phi_2 - 33\phi_1^2 - 87\phi_1\phi_2^2 + 21\phi_1\phi_2 + 22\phi_1 - 59\phi_2^3 + 13\phi_2^2 + 50\phi_2 + 5) \geq \lambda \\
& (\phi_1 + 2)(6\phi_1^2 + \phi_1\phi_2 - 11\phi_1 - 15\phi_2^2 + 16\phi_2 - 1) \geq \lambda \\
& (\phi_1 + 2)(-3\phi_1^2 - 8\phi_1\phi_2 + \phi_1 - 5\phi_2^2 - 5\phi_2 + 11) \geq \lambda \\
& (\phi_1 + 2)(-9\phi_1 + 13\phi_2 - \phi_1\phi_2 - 21\phi_2^2 + 12) \geq \lambda \\
& \phi_1 + \phi_2 \leq 1 \\
& \phi_1, \phi_2 \geq 0
\end{array}$$

$NLP_D(\mathcal{B}_0^{5.6})$:

$$\begin{array}{lc}
\max\limits_{\lambda,\phi,\sigma} & \lambda \\
\text{s.t.} & -\sigma - 1 \geq \lambda \\
& \phi - \sigma - 1 \geq \lambda \\
& -18\sigma - 34 \geq \lambda \\
& -9\sigma - 17 \geq \lambda \\
& \phi \in [-3, 1], \sigma \in [-3, 1]
\end{array}$$

The respective optimal solutions of $NLP_D(\mathcal{B}_0^{5.5})$ and $NLP_D(\mathcal{B}_0^{5.6})$ are approximately $(\lambda^*, \phi_1^*, \phi_2^*) = (6.5333, 0, 0.5333)$ and $(\lambda^{**}, \phi^{**}, \sigma^{**}) = (2, 0.5345, -3)$. Hence, by Proposition 5.65, $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$ and $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ are both full dimensional. Thus, Proposition 5.65, together with the procedures of Section 5.5, allows for the completion of the intialization phase of Outline 5.62. The next task we address is that of step 2: "Determine the set of feasible bases which have invariancy regions that are: (i) adjacent to $\mathcal{IR}_{\mathcal{B}}$, and (ii) at least $(k-1)$–dimensional."

This can be done by finding the set of $j \in \mathcal{B}$ such that $dim(\hbar^j_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}) = k-1$ and then, for each such index j, determining all bases which have invariancy regions that are adjacent to $\mathcal{IR}_{\mathcal{B}}$ along $\hbar^j_{\mathcal{B}}$. Consider the following observations, propositions, and definitions.

Observation 5.66. Let a complementary basis $\mathcal{B}$ and an index $i \in \mathcal{B}$ be given. It is clear from (5.82) and (5.84) that if $i \in (\mathrm{Z}_{\mathcal{B}} \cup \mathrm{E}_{\mathcal{B}})$ then $\hbar^i_{\mathcal{B}}$ does not form a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$.

Proposition 5.67. For a given complementary basis $\mathcal{B}$, if there exists $i \in (\mathcal{B} \setminus (\mathrm{Z}_{\mathcal{B}} \cup \mathrm{E}_{\mathcal{B}}))$, $\phi' \in S_\phi$ and $\sigma' \in S_\sigma$ such that $g(\mathcal{B})\left(Adj(G(\phi')_{\bullet\mathcal{B}})\right)_{j\bullet} q(\phi',\sigma') > 0$ for all $j \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right)$ and $\left(Adj(G(\phi')_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi',\sigma') = 0$, then $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$. Hence, if the following NLP has a strictly positive optimal value, $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$.

$$\begin{array}{rll} NLP_F(\mathcal{B}, i) := & \max\limits_{\lambda,\phi,\sigma} & \lambda \\ & \text{s.t.} & g(\mathcal{B})\left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{j\bullet} q(\phi,\sigma) \geq \lambda \mathbf{1} \quad \forall\, j \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right) \\ & & \left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi,\sigma) = 0 \\ & & \phi \in S_\phi, \ \sigma \in S_\sigma \end{array} \tag{5.88}$$

Proof. Suppose $(\lambda^*, \phi^*, \sigma^*)$ is a solution to $NLP_F(\mathcal{B}, i)$ and $\lambda^* > 0$. Recognize that since $(\lambda^*, \phi^*, \sigma^*)$ is feasible to $NLP_F(\mathcal{B}, i)$, we have $\left(Adj(G(\phi^*)_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi^*,\sigma^*) = 0$ which clearly shows that $\theta^* = (\phi^*, \sigma^*)$ lies on the hypersurface $\hbar^i_{\mathcal{B}}$. We proceed by showing that $\theta^* \in \mathcal{IR}_{\mathcal{B}}$ and that the defining constraints of $\mathcal{IR}_{\mathcal{B}}$ which are implied by $\hbar^i_{\mathcal{B}}$ cannot be removed without adding new points to $\mathcal{IR}_{\mathcal{B}}$, i.e., $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$.

We now show that $\theta^* \in \mathcal{IR}_{\mathcal{B}}$ by showing that θ^* satisfies all defining constraints of $\mathcal{IR}_{\mathcal{B}}$. Recognize that since θ^* satifies the constraints of $NLP_F(\mathcal{B}, i)$, we have that θ^* satisfies the defining constraints of $\mathcal{IR}_{\mathcal{B}}$ associated with each element of $\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}}\right)$. From (5.85) notice that for each $\ell \in \mathcal{B}$, if $\ell \in \mathrm{H}^i_{\mathcal{B}}$ then $(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}) \subseteq (\hbar^\ell_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}})$. Hence, since $(Adj(G(\phi^*)_{\bullet\mathcal{B}}))_{i\bullet} q(\phi^*,\sigma^*) = 0$, it is clear from (5.83) that $(Adj(G(\phi^*)_{\bullet\mathcal{B}}))_{\ell\bullet} q(\phi^*,\sigma^*) = 0$ for all $\ell \in \mathrm{H}^i_{\mathcal{B}}$. This also shows that θ^* satisfies all defining constraints of $\mathcal{IR}_{\mathcal{B}}$ associated with elements of $\mathrm{H}^i_{\mathcal{B}}$. Next, recall from (5.82) that for all $j \in \mathrm{Z}_{\mathcal{B}}$, $\left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{j\bullet} q(\phi,\sigma) = 0$ for all $\phi \in S_\phi$ and $\sigma \in S_\sigma$. Therefore θ^* trivially satisfies all defining constraints of $\mathcal{IR}_{\mathcal{B}}$ associated with elements of $\mathrm{Z}_{\mathcal{B}}$. Thus, we now have that

θ^* satisfies all defining constraints of $\mathcal{IR}_{\mathcal{B}}$. Furthermore, since $\lambda^* > 0$, there must exist $\epsilon > 0$ such that all $\theta = (\phi, \sigma)$ in the ball of radius ϵ centered at θ^* satisfy $(Adj(G(\phi)_{\bullet\mathcal{B}}))_{j\bullet}q(\phi,\sigma) \geq 0$ for all $j \in \left(\mathcal{B} \setminus \left(\mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right)$. Hence, since θ^* lies on $\hbar^i_{\mathcal{B}}$, this ϵ-ball must contain a $\hat{\theta}$ satisfying all the defining inequalities of $\mathcal{IR}_{\mathcal{B}}$ except those implied by $\hbar^i_{\mathcal{B}}$. Thus, the inequalities implied by $\hbar^i_{\mathcal{B}}$ cannot be removed from the description of $\mathcal{IR}_{\mathcal{B}}$ without altering its structure and therefore $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$. □

Observation 5.68. The converse of Proposition 5.67 also holds if $\mathcal{IR}_{\mathcal{B}}$ is full dimensional.

Observation 5.69. Let a complementary basis $\mathcal{B}$ and an index $i \in \mathcal{B}$ be given such that $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$. It is clear from (5.85) that if $j \in \mathrm{H}^i_{\mathcal{B}}$, $\hbar^j_{\mathcal{B}}$ also forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$.

Together, Observations 5.66, 5.68 and 5.69 and Proposition 5.67 provide us with a strategy so that, given a complementary basis $\mathcal{B}$ such that $\mathcal{IR}_{\mathcal{B}}$ is full dimensional, we can compute the set

$$\mathrm{F}_{\mathcal{B}} := \left\{ i \in \mathcal{B} : \hbar^i_{\mathcal{B}} \text{ forms a } (k-1)\text{-dimensional boundary of } \mathcal{IR}_{\mathcal{B}} \right\} \tag{5.89}$$

of indices in $\mathcal{B}$ whose associated hypersurfaces form $(k-1)$-dimensional boundaries of $\mathcal{IR}_{\mathcal{B}}$. We present this strategy in Algorithm 5.2.

Algorithm 5.2 BuildF($\mathcal{B}$) – Build $\mathrm{F}_{\mathcal{B}}$.
Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}_{\mathcal{B}}) = k$. (Assume that the sets $\mathrm{Z}_{\mathcal{B}}, \mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^i_{\mathcal{B}}$ for all $i \in \mathcal{B}$ have been constructed.)
Output: The set $\mathrm{F}_{\mathcal{B}}$.

```
Let $\mathrm{F}_{\mathcal{B}} = \emptyset$.
for $i \in (\mathcal{B} \setminus (\mathrm{Z}_{\mathcal{B}} \cup \mathrm{E}_{\mathcal{B}} \cup \mathrm{F}_{\mathcal{B}}))$ do solve $NLP_F(\mathcal{B}, i)$ to find optimal solution $(\lambda^*, \phi^*, \sigma^*)$.
    if $\lambda^* > 0$ then add $\left(i \cup \mathrm{H}^i_{\mathcal{B}}\right)$ to $\mathrm{F}_{\mathcal{B}}$.
Return $\mathrm{F}_{\mathcal{B}}$.
```

We now return to our consideration of Examples 5.5 and 5.6. Recognize from (5.88) that in order to use BuildF (outlined in Algorithm 5.2) to create $F_{\mathcal{B}_0^{5.5}}$ and $F_{\mathcal{B}_0^{5.6}}$, we must first construct $\mathrm{E}_{\mathcal{B}_0^{5.5}}$, $\mathrm{H}^i_{\mathcal{B}_0^{5.5}}$ for each $i \in \mathcal{B}_0^{5.5}$, $\mathrm{E}_{\mathcal{B}_0^{5.6}}$ and $\mathrm{H}^j_{\mathcal{B}_0^{5.6}}$ for each $j \in \mathcal{B}_0^{5.6}$. For this we use the procedure

BuildEandH, as outlined in Algorithm 5.1. Observe $NLP_H(\mathscr{B}_0^{5.5}, w_1, z_2)$ (5.86), which we use for the first step of BuildH$(\mathscr{B}_0^{5.5})$:

$$\begin{array}{rc}
\max\limits_{\lambda,\phi} & \lambda \\
\text{s.t.} & -3\phi_1^3 - 18\phi_1^2\phi_2 + 49\phi_1^2 + 75\phi_1\phi_2^2 - 148\phi_1\phi_2 - 68\phi_1 - 96\phi_2^2 + 16\phi_2 + 76 = 0 \\
& (\phi_1 + 2)(9\phi_1^3 - 9\phi_1^2\phi_2 - 33\phi_1^2 - 87\phi_1\phi_2^2 + 21\phi_1\phi_2 + 22\phi_1 - 59\phi_2^3 + 13\phi_2^2 + 50\phi_2 + 5) \geq \lambda \\
& (\phi_1 + 2)(6\phi_1^2 + \phi_1\phi_2 - 11\phi_1 - 15\phi_2^2 + 16\phi_2 - 1) \geq 0 \\
& (\phi_1 + 2)(-3\phi_1^2 - 8\phi_1\phi_2 + \phi_1 - 5\phi_2^2 - 5\phi_2 + 11) \geq 0 \\
& (\phi_1 + 2)(-9\phi_1 + 13\phi_2 - \phi_1\phi_2 - 21\phi_2^2 + 12) \geq 0 \\
& \phi_1 + \phi_2 \leq 1 \\
& \phi_1, \phi_2 \geq 0
\end{array}$$

The approximate optimal solution to $NLP_H(\mathscr{B}_0^{5.5}, w_1, z_2)$ is $(\lambda^*, \phi^*, \sigma^*) = (34.8687, 0.1478, 0.8522)$ which, based on the result of Proposition 5.63, shows that $z_2 \notin \mathrm{H}^{w_1}_{\mathscr{B}_0^{5.5}}$. For the sake of space, we do not explicitly show the additional ten NLPs solved during the BuildH$(\mathscr{B}_0^{5.5})$ procedure, but we do provide the following outline of the results:

w_1: z_2 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, w_1, z_2) \approx 34.8687$. Do not add z_2 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ or $\mathrm{H}^{w_1}_{\mathscr{B}_0^{5.5}}$.

z_3 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, w_1, z_3) \approx 0.8015$. Do not add z_3 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ or $\mathrm{H}^{w_1}_{\mathscr{B}_0^{5.5}}$.

z_4 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, w_1, z_4) \approx 4.6874$. Do not add z_4 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ or $\mathrm{H}^{w_1}_{\mathscr{B}_0^{5.5}}$.

z_5 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, w_1, z_5) \approx 16.1191$. Do not add z_5 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ or $\mathrm{H}^{w_1}_{\mathscr{B}_0^{5.5}}$.

z_2: w_1 – $NLP_H(\mathscr{B}_0^{5.5}, z_2, w_1)$ is infeasible. Add z_2 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ and cease consideration of z_2.

z_3: w_1 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, z_3, w_1) \approx 76.64$. Do not add w_1 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ or $\mathrm{H}^{z_3}_{\mathscr{B}_0^{5.5}}$.

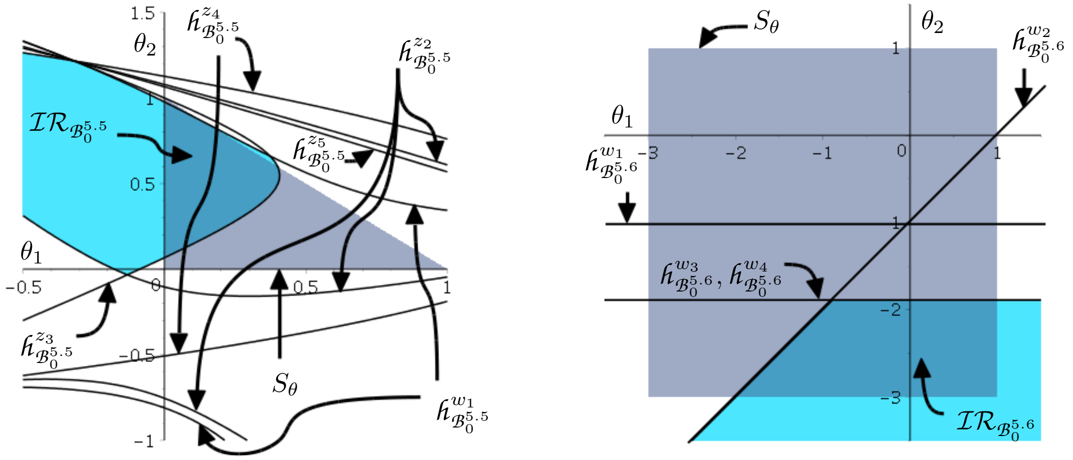
z_4 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, z_3, z_4) \approx 21.348$. Do not add z_4 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ or $\mathrm{H}^{z_3}_{\mathscr{B}_0^{5.5}}$.

z_5 – Optimal value of $NLP_H(\mathscr{B}_0^{5.5}, z_3, z_5) \approx 30.1402$. Do not add z_5 to $\mathrm{E}_{\mathscr{B}}$ or $\mathrm{H}^{z_3}_{\mathscr{B}_0^{5.5}}$.

z_4: w_1 – $NLP_H(\mathscr{B}_0^{5.5}, z_4, w_1)$ is infeasible. Add z_4 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ and cease consideration of z_4.

z_5: w_1 – $NLP_H(\mathscr{B}_0^{5.5}, z_5, w_1)$ is infeasible. Add z_5 to $\mathrm{E}_{\mathscr{B}_0^{5.5}}$ and cease consideration of z_5.

After running BuildEandH$(\mathscr{B}_0^{5.5})$ we find that $\mathrm{E}_{\mathscr{B}_0^{5.5}} = \{z_2, z_4, z_5\}$ and $\mathrm{H}^{i}_{\mathscr{B}_0^{5.5}} = \emptyset$ for all $i \in \mathscr{B}$. We do not explicitly show the steps of running BuildEandH$(\mathscr{B}_0^{5.6})$, but the results reveal that $\mathrm{E}_{\mathscr{B}_0^{5.6}} = \{w_1\}$, $\mathrm{H}^{w_1}_{\mathscr{B}_0^{5.5}} = \mathrm{H}^{w_2}_{\mathscr{B}_0^{5.5}} = \emptyset$, $\mathrm{H}^{w_3}_{\mathscr{B}_0^{5.5}} = \{w_4\}$, and $\mathrm{H}^{w_4}_{\mathscr{B}_0^{5.5}} = \{w_3\}$. One can observe from Figures 5.3a and 5.3b that these sets have been constructed correctly.



(a) Hypersurfaces for $\mathcal{B}_0^{5.5}$.

(b) Hypersurfaces for $\mathcal{B}_0^{5.6}$.

Figure 5.3: Hypersurfaces associated with initial bases for the two examples.

We are now ready to use the procedure BUILDF to construct the sets $\mathrm{F}_{\mathcal{B}_0^{5.5}}$ and $\mathrm{F}_{\mathcal{B}_0^{5.6}}$. For both examples, we first consider the index w_1. Recall, however, that $w_1 \in \mathrm{E}_{\mathcal{B}_0^{5.6}}$ and thus, by Observation 5.66, we can immediately conclude that $w_1 \notin \mathrm{F}_{\mathcal{B}_0^{5.6}}$. Hence, we need only observe $NLP_F(\mathcal{B}_0^{5.5}, w_1)$ (5.88):

$$
\begin{array}{ll}
\max\limits_{\lambda,\phi} & \lambda \\
\text{s.t.} & -3\phi_1^3 - 18\phi_1^2\phi_2 + 49\phi_1^2 + 75\phi_1\phi_2^2 - 148\phi_1\phi_2 - 68\phi_1 - 96\phi_2^2 + 16\phi_2 + 76 = 0 \\
& (\phi_1+2)(9\phi_1^3 - 9\phi_1^2\phi_2 - 33\phi_1^2 - 87\phi_1\phi_2^2 + 21\phi_1\phi_2 + 22\phi_1 - 59\phi_2^3 + 13\phi_2^2 + 50\phi_2 + 5) \geq \lambda \\
& (\phi_1+2)(6\phi_1^2 + \phi_1\phi_2 - 11\phi_1 - 15\phi_2^2 + 16\phi_2 - 1) \geq \lambda \\
& (\phi_1+2)(-3\phi_1^2 - 8\phi_1\phi_2 + \phi_1 - 5\phi_2^2 - 5\phi_2 + 11) \geq \lambda \\
& (\phi_1+2)(-9\phi_1 + 13\phi_2 - \phi_1\phi_2 - 21\phi_2^2 + 12) \geq \lambda \\
& \phi_1 + \phi_2 \leq 1 \\
& \phi_1, \phi_2 \geq 0
\end{array}
$$

The approximate optimal solution is $(\lambda^*, \phi_1^*, \phi_2^*) = (0.8015, 0.1388, 0.8599)$. Thus, by Proposition 5.67 and using (5.83), we see that $\hbar^{w_1}_{\mathcal{B}_0^{5.5}} = \{\phi \in S_\phi : -3\phi_1^3 - 18\phi_1^2\phi_2 + 49\phi_1^2 + 75\phi_1\phi_2^2 - 148\phi_1\phi_2 - 68\phi_1 - 96\phi_2^2 + 16\phi_2 + 76 = 0\}$ forms a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$. By carrying out the remainder of the procedures BUILDF$\left(\mathcal{B}_0^{5.5}\right)$ and BUILDF$\left(\mathcal{B}_0^{5.6}\right)$, we find that $\hbar^{z_3}_{\mathcal{B}_0^{5.5}} = \{\phi \in S_\phi : (\phi_1+2)(6\phi_1^2 + \phi_1\phi_2 - 11\phi_1 - 15\phi_2^2 + 16\phi_2 - 1) = 0\}$ forms the only additional $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$ and $\hbar^{w_2}_{\mathcal{B}_0^{5.6}} = \{\theta \in S_\theta : \phi - \sigma - 1 = 0, \theta = (\phi, \sigma)\}$ and $\hbar^{w_3}_{\mathcal{B}_0^{5.6}} = \hbar^{w_4}_{\mathcal{B}_0^{5.6}} = \{\theta \in S_\theta : -9\sigma - 17 = 0, \phi \in S_\phi, \theta = (\phi, \sigma)\}$ form the $(k-1)$-dimensional boundaries of $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$. To see that the correct conclusions have been made about which surfaces form $(k-1)$-

dimensional boundaries of $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$ and $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$, observe Figure 5.3 which shows $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$, $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ and the hypersurfaces associated with each basic variable for Examples 5.5 and 5.6.

We now continue our discussion and develop the tools necessary for determining bases having invariancy regions which are adjacent to a given invariancy region across a $(k-1)$-dimensional boundary. Consider the following definition, lemma, and proposition.

Definition 5.70. Given a complementary basis $\mathcal{B}$, the associated tableau $T_{\mathcal{B}}(\phi)$, and distinct indices $i \in \mathcal{B}$ and $j \in \mathcal{E}$, a *pivot on* $(T_{\mathcal{B}}(\phi))_{i,j}$ is the process of creating a new matrix T^* by performing elementary row operations on $T_{\mathcal{B}}(\phi)$ so that $T^*_{i,j} = 1$ and $T^*_{\gamma,j} = 0$ for all $\gamma \in (\mathcal{B} \setminus \{i\})$.

Note that for any basis $\mathcal{B}$ a pivot on $(T_{\mathcal{B}}(\phi))_{i,j}$ can only be made if $(T_{\mathcal{B}}(\phi))_{i,j}$ is not identically zero.

Lemma 5.71. Let a feasible basis $\mathcal{B}$ be given along with distinct $i, j \in \mathcal{B}$. If $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is identically zero and there exists $\phi' \in \mathcal{IR}^{\phi}_{\mathcal{B}}$ such that $(T_{\mathcal{B}}(\phi'))_{j,\bar{\imath}} > 0$ and $(T_{\mathcal{B}}(\phi'))_{i,\bar{\jmath}} \neq 0$, then $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$ for all $\phi \in S_{\phi}$ and $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} \neq 0$ for all $\phi \in S_{\phi}$.

Proof. By Proposition 5.61, $\mathcal{B}' = (\mathcal{B} \setminus \{i,j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a feasible complementary basis. Since $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is identically zero in this case, the tableau $T_{\mathcal{B}'}(\phi)$ can be obtained from the tableau $T_{\mathcal{B}}$ in two steps: (i) create matrix T^* from $T_{\mathcal{B}}(\phi)$ by performing a pivot on $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}$, and (ii) obtain $T_{\mathcal{B}'}(\phi)$ from T^* by performing a pivot on $(T^*)_{i,\bar{\jmath}}$. Consider the following subset of $T_{\mathcal{B}}(\phi)$:

	i	j	$\bar{\imath}$	$\bar{\jmath}$	
i	1	0	0	$(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}}$	$\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)$
γ	0	0	$(T_{\mathcal{B}}(\phi))_{\gamma,\bar{\imath}}$	$(T_{\mathcal{B}}(\phi))_{\gamma,\bar{\jmath}}$	$\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet} q(\phi,\sigma)$
j	0	1	$(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}$	$(T_{\mathcal{B}}(\phi))_{j,\bar{\jmath}}$	$\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} q(\phi,\sigma)$

where γ represents any element of $\mathcal{B} \setminus \{i,j\}$. Then, by pivoting on $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}$, we obtain the following corresponding subset of T^*:

	i	j	$\bar{\imath}$	$\bar{\jmath}$	
i	1	0	0	$(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}}$	$\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)$
γ	0	$-\frac{(T_{\mathcal{B}}(\phi))_{\gamma,\bar{\imath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}$	0	$(T_{\mathcal{B}}(\phi))_{\gamma,\bar{\jmath}} - (T_{\mathcal{B}}(\phi))_{\gamma,\bar{\imath}} \frac{(T_{\mathcal{B}}(\phi))_{j,\bar{\jmath}}}{(T_{\mathcal{B}})_{j,\bar{\imath}}}$	$\left(\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{\gamma\bullet} - \frac{(T_{\mathcal{B}}(\phi))_{\gamma,\bar{\imath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet}\right) q(\phi,\sigma)$
$\bar{\imath}$	0	$\frac{1}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}$	1	$\frac{(T_{\mathcal{B}}(\phi))_{j,\bar{\jmath}}}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}$	$\frac{\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} q(\phi,\sigma)}{(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}$

Finally, $T_{\mathcal{B}'}(\phi)$ can be obtained by pivoting on $(T^*)_{i,\bar{\jmath}}$. Observe the following elements of interest from the RHS of $T_{\mathcal{B}'}(\phi)$:

$\bar{j}$	$\dfrac{\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)}{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}}$
$\bar{\imath}$	$\dfrac{1}{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}\Big((T_{\mathcal{B}}(\phi))_{i,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} - (T_{\mathcal{B}}(\phi))_{j,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\Big) q(\phi,\sigma)$

(5.90)

Recall that the RHS of the tableau for an arbitrary basis $\mathcal{B}^*$ is $G(\phi)^{-1}_{\bullet\mathcal{B}^*}q(\phi,\sigma) = \dfrac{Adj(G(\phi)_{\bullet\mathcal{B}^*})}{det(G(\phi)_{\bullet\mathcal{B}^*})}q(\phi,\sigma)$. Hence, the tableau above shows that

$$\frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{j}\bullet}}{det(G(\phi)_{\bullet\mathcal{B}'})}q(\phi,\sigma) = \frac{\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)}{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}} \tag{5.91}$$

and

$$\frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{\imath}\bullet}}{det(G(\phi)_{\bullet\mathcal{B}'})}q(\phi,\sigma) = \\ \frac{1}{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}\Big((T_{\mathcal{B}}(\phi))_{i,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} - (T_{\mathcal{B}}(\phi))_{j,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\Big) q(\phi,\sigma). \tag{5.92}$$

Consider (5.91) and observe the following:

$$\begin{aligned}
\frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{j}\bullet}}{det(G(\phi)_{\bullet\mathcal{B}'})}q(\phi,\sigma) &= \frac{\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)}{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}} \\
&= \frac{(Adj(G(\phi)_{\bullet\mathcal{B}}))_{i\bullet} q(\phi,\sigma)}{det(G(\phi)_{\bullet\mathcal{B}})(T_{\mathcal{B}}(\phi))_{i,\bar{j}}}
\end{aligned}$$

$$\Rightarrow \qquad det(G(\phi)_{\bullet\mathcal{B}'}) = \frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{j}\bullet} q(\phi,\sigma)}{(Adj(G(\phi)_{\bullet\mathcal{B}}))_{i\bullet} q(\phi,\sigma)} det(G(\phi)_{\bullet\mathcal{B}})(T_{\mathcal{B}}(\phi))_{i,\bar{j}}$$

Hence, from Lemma 5.15 we can conclude that either $(T_{\mathcal{B}}(\phi))_{i,\bar{j}} > 0$ for all $\phi \in S_\phi$ or $(T_{\mathcal{B}}(\phi))_{i,\bar{j}} < 0$ for all $\phi \in S_\phi$ and thus $(T_{\mathcal{B}}(\phi))_{i,\bar{j}} \neq 0$ for all $\phi \in S_\phi$. Next consider (5.92) and observe the following:

$$\frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{\imath}\bullet}}{det(G(\phi)_{\bullet\mathcal{B}'})}q(\phi,\sigma) = \frac{\Big((T_{\mathcal{B}}(\phi))_{i,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} - (T_{\mathcal{B}}(\phi))_{j,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet}\Big)}{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}q(\phi,\sigma)$$

$$\Rightarrow \frac{det(G(\phi)_{\bullet\mathcal{B}'})}{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{\imath}\bullet} q(\phi,\sigma)} = \frac{(T_{\mathcal{B}}(\phi))_{i,\bar{j}}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}{\Big((T_{\mathcal{B}}(\phi))_{i,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} q(\phi,\sigma) - (T_{\mathcal{B}}(\phi))_{j,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)\Big)}$$

$$\begin{aligned}
\Rightarrow det(G(\phi)_{\bullet\mathcal{B}'}) &= \frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{\imath}\bullet} q(\phi,\sigma)(T_{\mathcal{B}}(\phi))_{i,\bar{j}}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}}{\Big((T_{\mathcal{B}}(\phi))_{i,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{j\bullet} q(\phi,\sigma) - (T_{\mathcal{B}}(\phi))_{j,\bar{j}}\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)\Big)} \\
&= \frac{(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\bar{\imath}\bullet} q(\phi,\sigma)(T_{\mathcal{B}}(\phi))_{i,\bar{j}}(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}}\, det(G(\phi)_{\bullet\mathcal{B}})}{\Big((T_{\mathcal{B}}(\phi))_{i,\bar{j}}\left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{j\bullet} q(\phi,\sigma) - (T_{\mathcal{B}}(\phi))_{j,\bar{j}}\left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi,\sigma)\Big)}
\end{aligned} \tag{5.93}$$

Recall that each element of the tableau associated with any feasible complementary basis is a rational function whose numerator and denominator are both polynomials in $\theta = (\phi, \sigma)$ such that the denominator does not equal zero for any $\theta \in S_\theta$. Hence, we can conclude that every element of the tableau associated with any feasible complementary basis is continuous for all $\theta \in S_\theta$. Thus, we must have that the denominator of (5.93) has constant sign for all $\theta \in S_\theta$. We can therefore conclude from this fact, the fact that $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}}$ has constant sign for all $\phi \in S_\phi$, Lemma 5.15 and (5.93) that either $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$ for all $\phi \in S_\phi$ or $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} < 0$ for all $\phi \in S_\phi$. Therefore, since we have $(T_{\mathcal{B}}(\phi'))_{j,\bar{\imath}} > 0$, it must be that $(T_{\mathcal{B}}(\phi))_{j,\bar{\imath}} > 0$ for all $\phi \in S_\phi$. □

Proposition 5.72. Let $\mathcal{B}$ be a feasible basis such that $\mathcal{IR}_{\mathcal{B}}$ is full dimensional and let $\hbar^i_{\mathcal{B}}$ be a $(k-1)$-dimensional boundary of $\mathcal{IR}_{\mathcal{B}}$. For any complementary set $\mathcal{B}' \neq \mathcal{B}$ such that $|\mathcal{B} \cap \mathcal{B}'| \geq h-2$, $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ if and only if one of the following conditions holds.

1. $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ and $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero.
2. $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$, $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is identically zero, there exists $\phi' \in S_\phi$ such that $(T_{\mathcal{B}}(\phi'))_{j,\bar{\imath}} > 0$ and $(T_{\mathcal{B}}(\phi'))_{i,\bar{\jmath}} \neq 0$, and the following NLP has a strictly positive optimal value:

$$\begin{array}{lll} NLP_A(\mathcal{B}, i, j) := & & \\ \max\limits_{\lambda,\phi,\sigma} & \lambda & \\ \text{s.t.} & g(\mathcal{B})\,(Adj(G(\phi)_{\bullet\mathcal{B}}))_{\xi\bullet}\, q(\phi,\sigma) \geq \lambda \mathbf{1} & \forall\, \xi \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right) \\ & (Adj(G(\phi)_{\bullet\mathcal{B}}))_{i\bullet}\, q(\phi,\sigma) = 0 & \\ & g(\mathcal{B}')\,(Adj(G(\phi)_{\bullet\mathcal{B}'}))_{\xi\bullet}\, q(\phi,\sigma) \geq \lambda \mathbf{1} & \forall\, \xi \in \left(\mathcal{B}' \setminus \left(\mathrm{Z}_{\mathcal{B}'} \cup \mathrm{H}^{\bar{\jmath}}_{\mathcal{B}'} \cup \{\bar{\jmath}\}\right)\right) \\ & \phi \in S_\phi, \sigma \in S_\sigma & \end{array} \tag{5.94}$$

Proof. ($\Leftarrow$): Consider the two conditions:

<u>Condition 1</u>: $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ and $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero.

By Observation 5.57, $\mathcal{B}'$ is a basis. The adjacency of $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ along $\hbar^i_{\mathcal{B}}$ comes from the following facts: (i) $\left(\mathcal{AS}(\phi) \cap cone\left(G(\phi)^{-1}_{\bullet(\mathcal{B}\setminus\{i\})}\right)\right)$ forms a common boundary of both $\mathcal{ID}_{\mathcal{B}}$ and $\mathcal{ID}_{\mathcal{B}'}$, and (ii) $\mathcal{ID}_{\mathcal{B}}$ and $\mathcal{ID}_{\mathcal{B}'}$ share this common boundary if and only if $\hbar^i_{\mathcal{B}}$ is a common boundary of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$.

<u>Condition 2</u>: $\mathcal{B}' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$, $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is identically zero, and $NLP_A(\mathcal{B}, i, j)$ has a strictly positive optimal value.

By Proposition 5.61, $\mathcal{B}'$ is a basis. We will show that if there exists a solution $(\lambda^*, \phi^*, \sigma^*)$ to $NLP_A(\mathcal{B}, i, j)$ such that $\lambda^* > 0$, then $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of both $\mathcal{IR}_{\mathcal{B}}$ and

$\mathcal{IR}_{\mathcal{B}'}$, and furthermore, $dim\left(\left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right) \cap \left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}\right)\right) = k-1$. Observe from (5.90) that the RHS of $T_{\mathcal{B}'}(\phi)$ associated with index $\bar{\jmath}$ is $\dfrac{\left(G(\phi)^{-1}_{\bullet\mathcal{B}}\right)_{i\bullet} q(\phi,\sigma)}{(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}}}$. Also, recognize from Lemma 5.71 that $(T_{\mathcal{B}}(\phi))_{i,\bar{\jmath}} \neq 0$ for all $\phi \in S_\phi$. Thus we see from (5.83) that $\hbar^i_{\mathcal{B}} \subseteq \hbar^{\bar{\jmath}}_{\mathcal{B}'}$. This shows that for every $\ell \in \mathrm{H}^{\bar{\jmath}}_{\mathcal{B}'}$, $(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}) \subseteq (\hbar^{\ell}_{\mathcal{B}'} \cap \mathcal{IR}_{\mathcal{B}})$. Thus, since $(\lambda^*, \phi^*, \sigma^*)$ is feasible to NLP_A and $\lambda^* > 0$, we have that $\theta^* = (\phi^*, \sigma^*)$ lies on $\hbar^i_{\mathcal{B}} \subseteq \hbar^{\bar{\jmath}}_{\mathcal{B}'}$ and satisfies strictly all defining constraints of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ which are not implied by $\hbar^i_{\mathcal{B}}$. Hence, $\hbar^i_{\mathcal{B}}$ forms a $(k-1)$-dimensional boundary of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ and $dim\left(\left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right) \cap \left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}\right)\right) = k-1$.

($\Rightarrow$): Since $\mathcal{B}'$ is a complementary set such that $\mathcal{B}' \neq \mathcal{B}$, $|\mathcal{B} \cap \mathcal{B}'| \geq h-2$, and $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$, we must have one of the following two cases:

Case 1: $|\mathcal{B} \cap \mathcal{B}'| = h-1$

In this case $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ if and only if $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis. Additionally, $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis if and only if there exists $\phi \in S_\phi$ such that $(T(\phi)_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$. Hence, in this case Condition 1 is satisfied.

Case 2: $|\mathcal{B} \cap \mathcal{B}'| = h-2$

In this case, $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ cannot be a basis and therefore there cannot exist $\phi \in S_\phi$ such that $(T(\phi)_{\mathcal{B}})_{i,\bar{\imath}} \neq 0$, i.e., $(T(\phi)_{\mathcal{B}})_{i,\bar{\imath}}$ is identically zero. Furthermore, the fact that $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent along $\hbar^i_{\mathcal{B}}$ implies that there exists $j \in \mathcal{B}$ such that $\mathcal{B}' = (\mathcal{B} \setminus \{i,j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$ is a basis. Additionally, as we showed in the proof for the reverse direction, we have $\hbar^i_{\mathcal{B}} \subseteq \hbar^{\bar{\jmath}}_{\mathcal{B}'}$. Since $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent across $\hbar^i_{\mathcal{B}}$, we know $dim\left(\left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right) \cap \left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}\right)\right) = k-1$. Furthermore, recognize that we can only have $dim\left(\left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right) \cap \left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}\right)\right) = k-1$ if there exists a point $\theta' = (\phi', \sigma')$ which lies on $\left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}}\right) \cap \left(\hbar^i_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}\right)$ and at which all defining constraints of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ which are not identically zero and not implied by $\hbar^i_{\mathcal{B}}$ are satisfied strictly, i.e., $g(\mathcal{B})\,(Adj(G(\phi')_{\bullet\mathcal{B}}))_{\xi\bullet}\, q(\phi', \sigma') > 0$ for all $\xi \in \left(\mathcal{B} \setminus \left(\mathrm{Z}_{\mathcal{B}} \cup \mathrm{H}^i_{\mathcal{B}} \cup \{i\}\right)\right)$ and $g(\mathcal{B}')\,(Adj(G(\phi')_{\bullet\mathcal{B}'}))_{\xi\bullet}\, q(\phi', \sigma') > 0$ for all $\xi \in \left(\mathcal{B}' \setminus \left(\mathrm{Z}_{\mathcal{B}'} \cup \mathrm{H}^{\bar{\jmath}}_{\mathcal{B}'} \cup \{\bar{\jmath}\}\right)\right)$. Let λ' represent the minimum value of the LHS of each of these inequalities. Then clearly $\lambda' > 0$ and $(\lambda', \phi', \sigma')$ is feasible to NLP_A. Therefore the optimal value of NLP_A must be strictly positive. □

Notice that Condition 1 of Proposition 5.72 indicates situations in which diagonal pivots can be used to obtain new adjacent invariancy regions, while Condition 2 indicates situations in which exchange pivots can be used to obtain new adjacent invariancy regions. Hence, combining Lemma

5.71 and Proposition 5.72 leads us to a strategy so that, given a given full dimensional invariancy region, we can compute the set

$$\mathrm{A}_{\mathcal{B}} := \left\{\text{Complementary bases } \mathcal{B}' : \mathcal{IR}_{\mathcal{B}'} \text{ is adjacent to } \mathcal{IR}_{\mathcal{B}}\right\} \tag{5.95}$$

of all adjacent invariancy regions. We note, however, that since the algorithm we are developing for mpLCP is iterative, there may often be times in which, given a basis $\mathcal{B}$, the set $\mathrm{A}_{\mathcal{B}}$ contains bases which have previously been discovered and whose invariancy regions have already been explored. In this case it is unnecessary to explore these bases agian. Thus, we introduce the following set, which we dynamically update throughout our procedure:

$$\mathscr{B} := \{\text{Complementary bases } \mathcal{B}' : \mathcal{B}' \text{ has been discovered and processed}\} \tag{5.96}$$

Hence, we are typically not interested in generating $\mathrm{A}_{\mathcal{B}}$ entirely. Instead we need the subset

$$\overline{\mathrm{A}}_{\mathcal{B}} := \mathrm{A}_{\mathcal{B}} \setminus \mathscr{B}. \tag{5.97}$$

We present a strategy for obtaining this set in Algorithm 5.3.

Algorithm 5.3 FINDADJACENTFULL($\mathcal{B}$,$\mathscr{B}$) – Determine all previously undiscovered invariancy regions which are adjacent to a given full dimensional invariancy region.
Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}_{\mathcal{B}}) = k$ and the set $\mathscr{B}$ of previously discovered bases. (Assume that the sets $\mathrm{Z}_{\mathcal{B}}, \mathrm{E}_{\mathcal{B}}, \mathrm{H}^i_{\mathcal{B}}$ for all $i \in \mathcal{B}$, and $\mathrm{F}_{\mathcal{B}}$ have been constructed.).
Output: The set $\overline{\mathrm{A}}_{\mathcal{B}}$ and an updated version of $\mathscr{B}$.

```
Let $\overline{\mathrm{A}}_{\mathcal{B}} = \emptyset$.
Select an arbitrary $\phi' \in S_{\phi}$.
for $i \in \mathrm{F}_{\mathcal{B}}$ do
    Let $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$.
    if $\mathcal{B}' \notin \mathscr{B}$ and $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero then add $\mathcal{B}'$ to $\overline{\mathrm{A}}_{\mathcal{B}}$ and $\mathscr{B}$.
    else
        for $j \in \mathcal{B} \setminus \{i\}$ do
            Let $\mathcal{B}'' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$.
            if $\mathcal{B}'' \notin \mathscr{B}$, $(T_{\mathcal{B}}(\phi'))_{j,\bar{\imath}} > 0$ and $(T_{\mathcal{B}}(\phi'))_{i,\bar{\jmath}} \neq 0$ then
                Perform pivots to find $T_{\mathcal{B}''}(\phi)$, then build $\mathrm{Z}_{\mathcal{B}''}$ and run BUILDEANDH($\mathcal{B}''$).
                Solve $NLP_A(\mathcal{B}, i, j)$ to obtain optimal solution $(\lambda^*, \phi^*, \sigma^*)$.
                if $\lambda^* > 0$ then add $\mathcal{B}''$ to $\overline{\mathrm{A}}_{\mathcal{B}}$ and $\mathscr{B}$.
Return $\overline{\mathrm{A}}_{\mathcal{B}}$ and $\mathscr{B}$.
```

We now again consider Examples 5.5 and 5.6. However, in order to highlight certain key qualities present in each example, we now consider them separately, beginning with Example 5.5. Due to the size of the tableaux associated with this example, they cannot be displayed here, but are instead included in Section 5.A. We will now use FindAdjacentFull, outlined in Algorithm 5.3, to find all bases whose invariancy regions are adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$. During our previous consideration of Example 5.5 we found that $\hbar^{w_1}_{\mathcal{B}_0^{5.5}}$ and $\hbar^{z_3}_{\mathcal{B}_0^{5.5}}$ formed $(k-1)$-dimensional boundaries of $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$ and so $\mathrm{F}_{\mathcal{B}_0^{5.5}} = \{w_1, z_3\}$. Observe from Table 5.11 in Section 5.A that $(T(\phi)_{\mathcal{B}_0^{5.5}})_{w_1,z_1} = \frac{4\,(32\,\phi_2-7\,\phi_1+29)}{-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22}$ and $(T(\phi)_{\mathcal{B}_0^{5.5}})_{z_3,w_3} = \frac{-7\,\phi_1-8\,\phi_2+13}{-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22}$. Since neither of these elements are identically zero, by condition 1 of Proposition 5.72, performing diagonal pivots on these elements provides new bases $\mathcal{B}_1^{5.5} = \{z_1, z_2, z_3, z_4, z_5\}$ and $\mathcal{B}_2^{5.5} = \{w_1, z_2, w_3, z_4, z_5\}$ which have invariancy regions which are adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$. Thus, we add $\mathcal{B}_1^{5.5}$ and $\mathcal{B}_2^{5.5}$ to $\mathrm{A}_{\mathcal{B}_0^{5.5}}$ and conclude FindAdjacentFull. Note that tableaux for $\mathcal{B}_1^{5.5}$ and $\mathcal{B}_2^{5.5}$ are found in Tables 5.12 and 5.13, respectively, which are contained in Section 5.A. We now complete our consideration of Example 5.5. We note, however, that using NLP_D and the procedures BuildEandH and BuildF reveal that: (i) both $\mathcal{IR}_{\mathcal{B}_1^{5.5}}$ and $\mathcal{IR}_{\mathcal{B}_2^{5.5}}$ are full dimensional, (ii) no $(k-1)$-dimension boundaries other than $\hbar^{z_1}_{\mathcal{B}_1^{5.5}} = \hbar^{w_1}_{\mathcal{B}_0^{5.5}}$ exist for $\mathcal{IR}_{\mathcal{B}_1^{5.5}}$, and (iii) no $(k-1)$-dimension boundaries other than $\hbar^{w_3}_{\mathcal{B}_2^{5.5}} = \hbar^{z_3}_{\mathcal{B}_0^{5.5}}$ exist for $\mathcal{IR}_{\mathcal{B}_2^{5.5}}$. This shows that $\mathcal{IR}_{\mathcal{B}_0^{5.5}}$, $\mathcal{IR}_{\mathcal{B}_1^{5.5}}$ and $\mathcal{IR}_{\mathcal{B}_2^{5.5}}$ form a partition of S_θ for Example 5.5.

Let us now return our focus to Example 5.6. We use FindAdjacentFull to find all bases whose invariancy regions are adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$. During our previous consideration of Example 5.6 we found that $\hbar^{w_2}_{\mathcal{B}_0^{5.6}}$ and $\hbar^{w_3}_{\mathcal{B}_0^{5.6}} = \hbar^{w_4}_{\mathcal{B}_0^{5.6}}$ formed $(k-1)$-dimensional boundaries of $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ and so $\mathrm{F}_{\mathcal{B}_0^{5.6}} = \{w_2, w_3, w_4\}$. Hence, we would now like to find additional invariancy regions adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ along each of these boundaries. We begin by observing $(T(\phi)_{\mathcal{B}_0^{5.6}})_{w_2,z_2}$, $(T(\phi)_{\mathcal{B}_0^{5.6}})_{w_3,z_3}$ and $(T(\phi)_{\mathcal{B}_0^{5.6}})_{w_4,z_4}$ from Table 5.14 in Section 5.B. All are identically zero. This shows that condition 1 of Proposition 5.72 does not apply for any of the $(k-1)$-dimensional boundaries of $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ and so we must use condition 2. As we must check for pivots associated with the variables w_2, w_3 and w_4, we consider these variables one at a time, beginning with w_2. Observe the following vector, which is the z_2 column of $T(\phi)_{\mathcal{B}_0^{5.6}}$: $\begin{bmatrix} 0 \\ 0 \\ -3 \\ \phi+5 \end{bmatrix}$. Suppose that in Line 2 of Algorithm 5.3 we selected $\phi' = 0$.

Then we see that the condition of the "if" statement on Line 9 is only satisfied by w_4. Hence, we only consider the exchange pivot involving w_2 and w_4. Observe $NLP_A(\mathcal{B}_0^{5.6}, w_2, w_4)$ (5.94):

$$\begin{array}{cc}
\max\limits_{\lambda,\phi,\sigma} & \lambda \\
\text{s.t.} & -\sigma - 1 \geq \lambda \\
& -18\sigma - 34 \geq \lambda \\
& -9\sigma - 17 \geq \lambda \\
& \phi - \sigma - 1 = 0 \\
& -8\sigma - \phi\sigma - 8 \geq \lambda \\
& -((2\phi + 13)(9\sigma + 17)) \geq \lambda \\
& \phi \in [-3, 1], \sigma \in [-3, 1]
\end{array}$$

The optimal solution to $NLP_A(\mathcal{B}_0^{5.6}, w_2, w_4)$ is $(\lambda^*, \phi^*, \sigma^*) = (2, -2, -3)$, which shows that by Condition 2 of Proposition 5.72, the invariancy region associated with basis $\mathcal{B}_1^{5.6} = \{w_1, z_2, w_3, z_4\}$ is adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ across $\hbar^{w_2}_{\mathcal{B}_0^{5.6}}$. We therefore add $\mathcal{B}_1^{5.6}$ to $\overline{\mathrm{A}}_{\mathcal{B}_0^{5.6}}$ and $\mathscr{B}$. Observe the invariancy region $\mathcal{IR}_{\mathcal{B}_1^{5.6}} = \left\{ \theta = (\phi, \sigma) : \begin{array}{l} -(8\,\sigma + \phi\,\sigma + 8) \geq 0 \\ (\sigma - \phi + 1) \geq 0 \\ -\,(2\,\phi + 13)\ (9\,\sigma + 17) \geq 0 \\ -(9\,\sigma + 17) \geq 0 \end{array} \right\}$, which we derive from $T(\phi)_{\mathcal{B}_1^{5.6}}$ (Table 5.15 in Section 5.B).

Now consider w_3. It can be observed from $T(\phi)_{\mathcal{B}_0^{5.6}}$, found in Table 5.14 in Section 5.B, that due to the "if" statement on Line 9 of Algorithm 5.3, the only exchange pivots we must now consider are those involving: (i) w_3 and w_1, and (ii) w_3 and w_2. The respective optimal solutions to $NLP_A(\mathcal{B}_0^{5.6}, w_3, w_1)$ and $NLP_A(\mathcal{B}_0^{5.6}, w_3, w_2)$ are approximately $(-0.6667, 1, -1.8889)$ and $(0.8889, -1.6217, -1.8889)$. This shows that the basis $\{z_1, w_2, z_3, w_4\}$ does not yield an invariancy region which is adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ across $\hbar^{w_3}_{\mathcal{B}_0^{5.6}}$, but basis $\mathcal{B}_2^{5.6} = \{w_1, z_2, z_3, w_4\}$ does. We therefore add $\mathcal{B}_2^{5.6}$ to $\overline{\mathrm{A}}_{\mathcal{B}_0^{5.6}}$ and $\mathscr{B}$. Note that the tableau associated with $\mathcal{B}_2^{5.6}$ is found in Table 5.16 in Section 5.B. Observe $\mathcal{IR}_{\mathcal{B}_2^{5.6}}$:

$$\mathcal{IR}_{\mathcal{B}_2^{5.6}} = \left\{ \theta = (\phi, \sigma) : \begin{array}{l} -6\,\phi - 9\,\sigma - 9 \geq 0 \\ 3\,\phi - 3\,\sigma - 3 \geq 0 \\ 90\,\sigma + 170 \geq 0 \\ -5\ (2\,\phi + 13)\ (9\,\sigma + 17) \geq 0 \end{array} \right\} \tag{5.98}$$

Finally, consider w_4. The "if" statement on Line 9 of Algorithm 5.3 reveals that the only exchange pivot we must consider involves w_4 and w_1. The optimal solution to $NLP_A(\mathcal{B}_0^{5.6}, w_4, w_1)$ is approximately $(-0.3611, -1.25, -1.8889)$, which shows that basis $\{w_1, z_2, w_3, z_4\}$ does not yield an invariancy region adjacent to $\mathcal{IR}_{\mathcal{B}_0^{5.6}}$ across $\hbar^{w_4}_{\mathcal{B}_0^{5.6}}$. Although it may not be obvious, $\mathcal{IR}_{\mathcal{B}_2^{5.6}}$ is not full dimensional. For this reason we now again briefly pause our consideration of Example 5.6 in order to determine ways in which we can detect, and properly handle this type of situation.

Recall that, given a complementary basis $\mathcal{B}$ for which $dim(\mathcal{IR}_{\mathcal{B}}) = k$, we have now developed strategies for: (i) proving that $\mathcal{IR}_{\mathcal{B}}$ is full dimensional, (ii) discovering the $(k-1)$-dimensional boundaries of $\mathcal{IR}_{\mathcal{B}}$, and (iii) discovering additional invariancy regions which are adjacent to $\mathcal{IR}_{\mathcal{B}}$. Hence, the next set of procedures we still need in order to be able to partition S_θ are those which allow us to perform steps (ii) and (iii) above if $dim(\mathcal{IR}_{\mathcal{B}}) \neq k$. The following corollaries provide the theory necessary in order to perform these tasks.

Corollary 5.73. Let distinct feasible complementary bases $\mathcal{B}'$ and $\mathcal{B}$ be given which satisfy: (i) $dim(\mathcal{IR}_{\mathcal{B}}) = k$, (ii) $|\mathcal{B}' \cap \mathcal{B}| \geq h-2$, and (iii) $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\mathscr{h}^i_{\mathcal{B}}$. Then $dim\,(\mathcal{IR}_{\mathcal{B}'}) \geq k-1$. Furthermore, $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$ if and only if $|\mathrm{Z}_{\mathcal{B}'}| > h-(k-p)$ or the optimal value of $NLP_D(\mathcal{B}')$ is nonpositive.

Proof. First notice that $dim\,(\mathcal{IR}_{\mathcal{B}'}) \geq k-1$ since $\mathscr{h}^i_{\mathcal{B}}$ forms a $k-1$ dimensional facet of both $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$, as shown in the proof of Proposition 5.72. The fact that $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$ if and only if $|\mathrm{Z}_{\mathcal{B}'}| > h-(k-p)$ or the optimal value of $NLP_D(\mathcal{B}')$ is nonpositive follows directly from Proposition 5.65. □

Proposition 5.72 provides a strategy for determining the invariancy regions which are adjacent to a given full dimensional invariancy region. Corollary 5.73 provides a strategy for determining when a discovered invariancy region is $k-1$ dimensional. Suppose that for some complementary basis $\mathcal{B}$ we solve $NLP_D(\mathcal{B})$ and discover that $\mathcal{IR}_{\mathcal{B}}$ is not full dimensional. Recognize that Corollary 5.73 does not provide any insight into which indices in $\mathcal{B}$ ought to be pivoted on in order to yield a new basis $\mathcal{B}'$ for which $\mathcal{IR}_{\mathcal{B}'}$ is adjacent to $\mathcal{IR}_{\mathcal{B}}$ and at least $(k-1)$-dimensional. The following discussion addresses finding such indices. Consider the following proposition.

Proposition 5.74. Let distinct feasible complementary bases $\mathcal{B}'$ and $\mathcal{B}$ be given which satisfy: (i) $dim(\mathcal{IR}_{\mathcal{B}}) = k$, (ii) $|\mathcal{B}' \cap \mathcal{B}| \geq h-2$, and (iii) $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\mathscr{h}^i_{\mathcal{B}'}$. Then $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$ if and only if for every $\theta \in \mathcal{IR}_{\mathcal{B}'}$ there exists an $\epsilon_\theta > 0$ and an index $j \in \mathcal{B}' \setminus \{i\}$ such that the following hold:

(1) For all $\theta' = (\phi', \sigma') \in B_{\epsilon_\theta}(\theta) \cap \mathcal{IR}_{\mathcal{B}'}$, $(Adj(G(\phi')_{\bullet\mathcal{B}'})_{i,\bullet}\, q(\phi', \sigma') = (Adj(G(\phi')_{\bullet\mathcal{B}'})_{j,\bullet}\, q(\phi', \sigma')$.

(2) $B_{\epsilon_\theta}(\theta) \cap \{\theta = (\phi, \sigma) : g(\mathcal{B}')\,(Adj(G(\phi)_{\bullet\mathcal{B}'})_{i,\bullet}\, q(\phi, \sigma) > 0, g(\mathcal{B}')\,(Adj(G(\phi)_{\bullet\mathcal{B}'})_{j,\bullet}\, q(\phi, \sigma) > 0\} = \emptyset$.

Proof. ($\Rightarrow$): Since we know that $\mathcal{IR}_{\mathscr{B}'}$ is $(k-1)$-dimensional and adjacent to another invariancy region across $\hbar^i_{\mathscr{B}'}$, we know $\mathcal{IR}_{\mathscr{B}'} \subset \hbar^i_{\mathscr{B}'}$. Since there are a finite number of constraints defining $\mathcal{IR}_{\mathscr{B}'}$, we know that this can only be the case if for every $\theta \in \mathcal{IR}_{\mathscr{B}'}$, there exists $j \in \mathscr{B}' \setminus \{i\}$ such that: (i) $\hbar^i_{\mathscr{B}'}$ and $\hbar^j_{\mathscr{B}'}$ intersect at θ, (ii) $dim(\hbar^i_{\mathscr{B}'} \cap \hbar^j_{\mathscr{B}'} \cap \mathcal{IR}_{\mathscr{B}'}) = k-1$, and (iii) there exists $\epsilon_\theta > 0$ such that the intersections of the open semi-algebraic half-spaces $\{\theta = (\phi, \sigma) : g(\mathscr{B}')\,(Adj(G(\phi)_{\bullet\mathscr{B}'})_{i,\bullet}\, q(\phi, \sigma) > 0\}$ and $\{\theta = (\phi, \sigma) : g(\mathscr{B}')\,(Adj(G(\phi)_{\bullet\mathscr{B}'})_{j,\bullet}\, q(\phi, \sigma) > 0\}$ with $B_{\epsilon_\theta}(\theta)$ are disjoint. Recognize that points (i) and (ii) imply condition 1 and point (iii) implies condition 2.

($\Leftarrow$): Since we know that $\mathcal{IR}_{\mathscr{B}'}$ is adjacent to another invariancy region, from the definition of adjacency we know $dim(\mathcal{IR}_{\mathscr{B}'}) \geq k-1$. Recognize that condition 2 of the proposition implies that for every $\theta \in \mathcal{IR}_{\mathscr{B}'}$, there exists $j \in \mathscr{B}' \setminus \{i\}$ and $\epsilon_\theta > 0$ such that the open semi-algebraic half-spaces $\{\theta = (\phi, \sigma) : g(\mathscr{B}')\,(Adj(G(\phi)_{\bullet\mathscr{B}'})_{i,\bullet}\, q(\phi, \sigma) > 0\}$ and $\{\theta = (\phi, \sigma) : g(\mathscr{B}')\,(Adj(G(\phi)_{\bullet\mathscr{B}'})_{j,\bullet}\, q(\phi, \sigma) > 0\}$ are disjoint within $B_{\epsilon_\theta}(\theta)$. This is enough to show that $dim(\mathcal{IR}_{\mathscr{B}'}) \leq k-1$. Hence, we have $dim(\mathcal{IR}_{\mathscr{B}'}) = k-1$. □

The work done in the proof of Proposition 5.74 does provide some insight into which indices in a basis whose invariancy region is $(k-1)$-dimensional ought to be pivoted on, but in order to effectively use the result of the proposition, we require several concepts from algebra. For this we quote several definitions and theorems from Bôcher and Duval [13]. Many of the results we cite here are properties of polynomials of multiple variables. We note that although many of the cited results are given specifically for polynomials of three variables, as stated in [13], the concepts apply directly to polynomials in any number of variables and we therefore present the more general version of these results. We also note that the author of [13] uses the term *vanishes* to indicate the property of equalling zero.

Definition 5.75. (Definition 2 of Section 60 of Bôcher and Duval [13]) A polynomial is said to be *reducible* if it is identically equal to the product of two polynomials, neither of which is a constant.

Recognize that Definition 5.75 implies that a polynomial is *irreducible* if it cannot be written as the product of two nonconstant polynomials.

Proposition 5.76. (Shortened version of Theorem 6 of Section 76 of Bôcher and Duval [13]) A polynomial in k variables which is not identically zero can be resolved into the product of irreducible factors, no one of which is constant.

Proposition 5.77. (Theorem 8 of Section 76 of Bôcher and Duval [13]) If p and q are two polynomials in k variables which both vanish at the point $(x_1^0, \ldots, x_k^0)$ and of which q is irreducible, and if in the neighborhood N of $(x_1^0, \ldots, x_k^0)$ p vanishes at all points at which q vanishes, then q is a factor of p.

Definition 5.78. (Bôcher and Duval [13]) By the *greatest common divisor* of two polynomials is meant their common factor of greatest degree.

We now make the following important observations based on these definitions and propositions.

Observation 5.79. The converse of Proposition 5.77 holds if we remove the condition that the polynomial q be irreducible.

Observation 5.80. Suppose the polynomial q in Proposition 5.77 was not assumed to be irreducible, then the claim "q is a factor of p" can be replaced with "the greatest common divisor of q and p has a nonconstant factor which vanishes at $(x_1^0, \ldots, x_k^0)$."

Based on the results of Observation 5.79 and 5.80, we introduce some additional notation. Given a complementary basis $\mathcal{B}$ and distinct $i, j \in \mathcal{B}$, we define:

$$GCD(\mathcal{B}, i, j) := \text{Greatest common divisor of } \left(Adj(G(\phi)_{\bullet\mathcal{B}})_{i,\bullet}\, q(\phi,\sigma) \text{ and } \left(Adj(G(\phi)_{\bullet\mathcal{B}})_{j,\bullet}\, q(\phi,\sigma). \tag{5.99}$$

Then for each complementary basis $\mathcal{B}$ and each index $i \in \mathcal{B}$ we define the set $\mathrm{D}^i_{\mathcal{B}}$ so that:

$$\mathrm{D}^i_{\mathcal{B}} := \emptyset \text{ if } dim(\mathcal{IR}_{\mathcal{B}}) = k, \tag{5.100}$$

and

$$\mathrm{D}^i_{\mathcal{B}} := \big\{ j \in \mathcal{B} : GCD(\mathcal{B}, i, j) \text{ is a nonconstant polynomial and} \\ (\mathcal{IR}_{\mathcal{B}} \cap \hbar^i_{\mathcal{B}}) \subseteq (\mathcal{IR}_{\mathcal{B}} \cap GCD(\mathcal{B}, i, j)) \big\} \text{ otherwise.} \tag{5.101}$$

In order to clarify these notations and highlight their importance in solving mpLCP, we provide the following small example. Suppose we are given an instance of mpLCP in which $S_\theta = [-4, 4]^2$, and in the process of partitioning S_θ we discover a basis $\mathcal{B}$ with invariancy region

$$\mathcal{IR}_{\mathcal{B}} = \left\{ \theta \in S_\theta : \begin{array}{l} w_1 = (\theta_1 - 2)(\theta_2 + 3) \geq 0 \\ w_2 = (\theta_1 - 2)(\theta_3 - 3) \geq 0 \\ w_3 = -\theta_1 - \theta_2 + 4 \geq 0 \\ w_4 = \theta_1 + \theta_2 \geq 0 \end{array} \right\}.$$

This invariancy region is displayed in Figure 5.4.

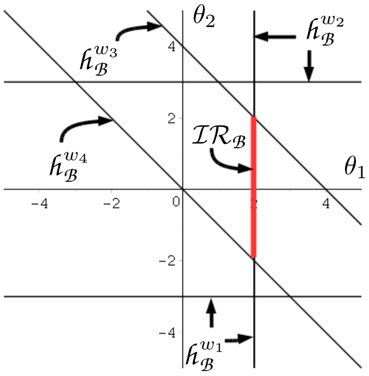

θ_2
$\hbar_{\mathcal{B}}^{w_2}$
$\hbar_{\mathcal{B}}^{w_3}$
4
2
$\mathcal{IR}_{\mathcal{B}}$
$\hbar_{\mathcal{B}}^{w_4}$
θ_1
-4
-2
0
2
4
-2
-4
$\hbar_{\mathcal{B}}^{w_1}$

Figure 5.4: Example of a $(k-1)$-dimensional region.

Recognize from Figure 5.4 that $dim(\mathcal{IR}_{\mathcal{B}}) = k-1$ even though there is no defining constraint of $\mathcal{IR}_{\mathcal{B}}$ for which the LHS is a constant multiple of another constraint's LHS. Examples such as this one motivate our use of the sets $GCD(\mathcal{B}, i, j)$ and $\mathrm{D}_{\mathcal{B}}^{i}$. Notice that in this example we have $GCD(\mathcal{B}, w_1, w_2) = \theta_1 - 2$ while the greatest common divisor for each other pair of variables is a constant. Further recognize that $\hbar_{\mathcal{B}}^{w_1} \cap \mathcal{IR}_{\mathcal{B}} = \hbar_{\mathcal{B}}^{w_2} \cap \mathcal{IR}_{\mathcal{B}}$. Hence, we can see that $\mathrm{D}_{\mathcal{B}}^{w_1} = \{w_2\}$, $\mathrm{D}_{\mathcal{B}}^{w_2} = \{w_1\}$, $\mathrm{D}_{\mathcal{B}}^{w_3} = \emptyset$ and $\mathrm{D}_{\mathcal{B}}^{w_4} = \emptyset$. This small example provides some insight into the types of situations in which $(k-1)$-dimensional regions can arise. In order to be able to study these situations further, we need to develop a method for constructing the set $\mathrm{D}_{\mathcal{B}}^{i}$ for a given basis $\mathcal{B}$ and index $i \in \mathcal{B}$. One strategy for doing this is to first compute $GCD(\mathcal{B}, i, j)$ for each $j \in \mathcal{B} \setminus \{i\}$ and then, if $GCD(\mathcal{B}, i, j)$ is a nonconstant polynomial, check to see if there is a point in $\mathcal{IR}_{\mathcal{B}}$ that lies on the surface $(Adj(G(\phi)_{\bullet\mathcal{B}}))_{i\bullet}\, q(\phi, \sigma) = 0$ but at which $GCD(\mathcal{B}, i, j)$ is nonzero. If no such point exists, include j in $\mathrm{D}_{\mathcal{B}}^{i}$. Otherwise, do not. Recognize that one way to determine whether such a point exists is to determine the maximum value of $GCD(\mathcal{B}, i, j)^2$ for points $\mathcal{IR}_{\mathcal{B}}$ which lie on the surface $(Adj(G(\phi)_{\bullet\mathcal{B}}))_{i\bullet}\, q(\phi, \sigma) = 0$. If this maximum value is zero, include j in $\mathrm{D}_{\mathcal{B}}^{i}$. Otherwise, do not. This strategy motivates the following proposition.

Proposition 5.81. Given complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}_{\mathcal{B}}) = k-1$ and an index $i \in \mathcal{B}$, we have that $j \in \mathcal{B} \setminus \{i\}$ is in $\mathrm{D}^i_{\mathcal{B}}$ if and only if $GCD(\mathcal{B}, i, j)$ is a nonconstant polynomial and the following NLP has an optimal value of zero:

$$\begin{array}{rcl} NLP_G(\mathcal{B}, i, j) := & \max\limits_{\lambda,\phi,\sigma} & \lambda \\ & \text{s.t.} & g(\mathcal{B})\left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{\ell\bullet} q(\phi,\sigma) \geq 0 \quad \forall\, \ell \in (\mathcal{B} \setminus (\mathrm{Z}_{\mathcal{B}} \cup \{i\})) \\ & & \left(Adj(G(\phi)_{\bullet\mathcal{B}})\right)_{i\bullet} q(\phi,\sigma) = 0 \\ & & GCD(\mathcal{B}, i, j)^2 \geq \lambda \\ & & \phi \in S_\phi, \ \sigma \in S_\sigma \end{array} \tag{5.102}$$

Proof. It is clear from (5.100) and (5.101) that j cannot be in $\mathrm{D}^i_{\mathcal{B}}$ unless $GCD(\mathcal{B}, i, j)$ is a nonconstant polynomial. Recognize the similarity between the second condition given in (5.101) and (5.85). Hence, the remainder of the proof is analogous to the proof of Proposition 5.63. □

The algebraic properties of polynomials that we have now introduced, together with the subsequent observations we have made and the result of Proposition 5.81, lead to the following corollary which plays an important role in handling $(k-1)$-dimensional invariancy regions.

Corollary 5.82. Let distinct feasible complementary bases $\mathcal{B}'$ and $\mathcal{B}$ be given which satisfy: (i) $dim(\mathcal{IR}_{\mathcal{B}}) = k$, (ii) $|\mathcal{B}' \cap \mathcal{B}| \geq h-2$, and (iii) $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\mathscr{h}^i_{\mathcal{B}'}$. Then $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$ if and only if $\mathrm{D}^i_{\mathcal{B}'} \neq \emptyset$.

Proof. The reverse direction of the proof follows directly from (5.100), (5.101) and the fact that since $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\mathscr{h}^i_{\mathcal{B}'}$ we know $dim(\mathcal{IR}_{\mathcal{B}'}) \geq k-1$. Hence, we focus on the forward direction and assume that $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$. Recall that because $dim\,(\mathcal{IR}_{\mathcal{B}'}) = k-1$ and $\mathcal{IR}_{\mathcal{B}}$ is adjacent to $\mathcal{IR}_{\mathcal{B}'}$ along $\mathscr{h}^i_{\mathcal{B}'}$, we have that for all $\theta = (\phi, \sigma) \in \mathcal{IR}_{\mathcal{B}'}$, $\left(Adj(G(\phi')_{\bullet\mathcal{B}'})\right)_{i,\bullet} q(\phi', \sigma') = 0$. Thus, using the vocabulary of [13], condition (1) of Proposition 5.74 implies that there exist $j \in \mathcal{B}' \setminus \{i\}$ and $\theta' \in \mathcal{IR}_{\mathcal{B}'}$ such that in a neighborhood around θ', $\left(Adj(G(\phi')_{\bullet\mathcal{B}'})\right)_{j,\bullet} q(\phi', \sigma')$ vanishes whenever $\left(Adj(G(\phi')_{\bullet\mathcal{B}'})\right)_{i,\bullet} q(\phi', \sigma')$ vanishes. From Proposition 5.77 and Observation 5.80 and using (5.99) we conclude that: (i) $GCD(\mathcal{B}', i, j)$ is a nonconstant polynomial, and (ii) $(\mathcal{IR}_{\mathcal{B}} \cap \mathscr{h}^i_{\mathcal{B}}) \subseteq (\mathcal{IR}_{\mathcal{B}} \cap GCD(\mathcal{B}, i, j))$. Hence, we observe from (5.101) that $\mathrm{D}^i_{\mathcal{B}'}$ will not be empty. □

The result of Corollary 5.82 now shows us that, given a complementary basis $\mathcal{B}$ for which $\mathcal{IR}_{\mathcal{B}}$ is $(k-1)$-dimensional and contained within $\mathscr{h}^i_{\mathcal{B}}$ for some $i \in \mathcal{B}$, the indices in $\mathcal{B}\setminus\{i\}$ which, when pivoted on, have the potential to yield new bases whose associated invariancy regions are adjacent to $\mathcal{IR}_{\mathscr{B}}$ and at least $(k-1)$-dimensional, are precisely those in $\mathrm{D}^i_{\mathscr{B}}$. One of the final theoretical result we need before being able to present a complete method for partitioning S_θ given an initial basis with a full dimensional invariancy region is a strategy for determining whether or not a pivot on an index in $\mathrm{D}^i_{\mathscr{B}}$ will yield an adjacent invariancy region. For this purpose we introduce the following proposition.

Proposition 5.83. Let distinct bases $\mathscr{B}$ and $\mathscr{B}'$ be given for which $dim(\mathcal{IR}_{\mathscr{B}}) = k-1$ and $|\mathscr{B} \cap \mathscr{B}'| \geq h-2$. Furthermore, let $\ell \in \mathscr{B}$ be the index for which we know $\mathcal{IR}_{\mathscr{B}} \subset \hslash^\ell_{\mathscr{B}}$. Then $\mathcal{IR}_{\mathscr{B}'}$ is adjacent to $\mathcal{IR}_{\mathscr{B}}$ along $\hslash^\ell_{\mathscr{B}}$ if and only if there exists $i \in \mathrm{D}^\ell_{\mathscr{B}}$ such that one of the following conditions holds:

1. $\mathscr{B}' = (\mathscr{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ and $(T_{\mathscr{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero.
2. There exists an additional index $j \in \mathscr{B} \setminus \{i\}$ such that $\mathscr{B}' = (\mathscr{B} \setminus \{i,j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$, $(T_{\mathscr{B}}(\phi))_{i,\bar{\imath}}$ is identically zero, there exists $\phi' \in S_\phi$ such that $(T_{\mathscr{B}}(\phi'))_{j,\bar{\imath}} > 0$ and $(T_{\mathscr{B}}(\phi'))_{i,\bar{\jmath}} \neq 0$, and the following NLP has a strictly positive optimal value:

$$NLP_{A2}(\mathscr{B}, i, j) := \begin{array}{lll} \max\limits_{\lambda,\phi,\sigma} & \lambda & \\ \text{s.t.} & g(\mathscr{B})\,(Adj(G(\phi)_{\bullet\mathscr{B}}))_{\xi\bullet}\,q(\phi,\sigma) \geq \lambda\mathbf{1} & \forall\, \xi \in \left(\mathscr{B} \setminus \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{H}^i_{\mathscr{B}} \cup \{i\}\right)\right) \\ & (Adj(G(\phi)_{\bullet\mathscr{B}}))_{i\bullet}\,q(\phi,\sigma) = 0 & \\ & g(\mathscr{B}')\,(Adj(G(\phi)_{\bullet\mathscr{B}'}))_{\xi\bullet}\,q(\phi,\sigma) \geq \lambda\mathbf{1} & \forall\, \xi \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'} \cup \mathrm{H}^{\bar{\jmath}}_{\mathscr{B}'} \cup \{\bar{\jmath}\}\right)\right) \\ & \phi \in S_\phi, \sigma \in S_\sigma & \end{array} \tag{5.103}$$

Proof. It is clear from Observation 5.80, (5.99), (5.100) and (5.101) that $dim(\mathcal{IR}_{\mathscr{B}} \cap \hslash^\ell_{\mathscr{B}} \cap \hslash^i_{\mathscr{B}}) = k-1$ if and only if $i \in \mathrm{D}^\ell_{\mathscr{B}}$. The remainder of the proof is analogous to that of Proposition 5.72. □

The result of Proposition 5.83 now provides us with a strategy so that, given a $(k-1)$-dimensional invariancy region $\mathscr{B}$, we can find the set $\overline{\mathrm{A}}_{\mathscr{B}}$ of all previously undiscovered invariancy regions whose invariancy regions are adjacent $\mathcal{IR}_{\mathscr{B}}$. We present this strategy in Algorithm 5.4.

Recognize the similarity of Algorithms 5.3 and 5.4. In fact, the "for" loop beginning on line 6 of Algorithm 5.4 follows exactly the same pattern as the "for" loop beginning on line 3 of Algorithm 5.3. The only difference is in the set of indices over which the loops iterate. This is simply due to the difference in the ways in which we detect indices whose associated hypersurfaces form $(k-1)$-dimensional boundaries of an invariancy region that is k, versus $(k-1)$, dimensional. In Algorithm 5.3 we deal with full dimensional regions, so the indices of interest are those in $\mathrm{F}_{\mathscr{B}}$, which we assume to be constructed prior to a call to Algorithm 5.3. In Algorithm 5.4 we deal with $(k-1)$ dimensional regions, and therefore, as indicated by Corollary 5.82 and Proposition 5.83, the indices

Algorithm 5.4 FindAdjacentKminus1($\mathcal{B}$,ℓ,$\mathscr{B}$) – Determine all previously undiscovered invariancy regions which are adjacent to a given $(k-1)$-dimensional invariancy region.
Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}_{\mathcal{B}}) = k-1$, an index $\ell \in \mathcal{B}$ for which we know $\mathcal{IR}_{\mathcal{B}} \subset \hbar^{\ell}_{\mathcal{B}}$ and the set $\mathscr{B}$ of previously discovered bases. (Assume that the sets $\mathrm{Z}_{\mathcal{B}}, \mathrm{E}_{\mathcal{B}}$, and $\mathrm{H}^{i}_{\mathcal{B}}$ for all $i \in \mathcal{B}$ have already been constructed.)
Output: The set $\overline{\mathrm{A}}_{\mathcal{B}}$ and an updated version of $\mathscr{B}$.

```
1: Let $\overline{\mathrm{A}}_{\mathcal{B}} = \emptyset$ and $\mathrm{D}^{\ell}_{\mathcal{B}} = \emptyset$.
2: Select an arbitrary $\phi' \in S_{\phi}$.
3: for $j \in \mathcal{B} \setminus \{\ell\}$ do compute $GCD(\mathcal{B}, \ell, j)$.
4:     if $GCD(\mathcal{B}, \ell, j)$ is a nonconstant polynomial then solve $NLP_G(\mathcal{B}, \ell, j)$ to obtain the optimal solution $(\lambda^*, \phi^*, \sigma^*)$.
5:         if $\lambda^* = 0$ then add $j$ to $\mathrm{D}^{\ell}_{\mathcal{B}}$.
6: for $i \in \mathrm{D}^{\ell}_{\mathcal{B}}$ do
7:     Let $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\overline{\imath}\}$
8:     if $\mathcal{B}' \notin \mathscr{B}$ and $(T_{\mathcal{B}}(\phi))_{i,\overline{\imath}}$ is not identically zero then add $\mathcal{B}'$ to $\overline{\mathrm{A}}_{\mathcal{B}}$ and $\mathscr{B}$.
9:     else
10:        for $j \in \mathcal{B} \setminus \{i\}$ do
11:            Let $\mathcal{B}'' = (\mathcal{B} \setminus \{i, j\}) \cup \{\overline{\imath}, \overline{\jmath}\}$
12:            if $\mathcal{B}'' \notin \mathscr{B}$, $(T_{\mathcal{B}}(\phi'))_{j,\overline{\imath}} > 0$ and $(T_{\mathcal{B}}(\phi'))_{i,\overline{\jmath}} \neq 0$ then
13:                Perform pivots to find $T_{\mathcal{B}''}(\phi)$, then build $\mathrm{Z}_{\mathcal{B}''}$ and run BuildEandH($\mathcal{B}''$).
14:                Solve $NLP_{A2}(\mathcal{B}, i, j)$ to obtain optimal solution $(\lambda^*, \phi^*, \sigma^*)$.
15:                if $\lambda^* > 0$ then add $\mathcal{B}''$ to $\overline{\mathrm{A}}_{\mathcal{B}}$ and $\mathscr{B}$.
16: Return $\overline{\mathrm{A}}_{\mathcal{B}}$ and $\mathscr{B}$.
```

of interest are those in $\mathrm{D}^{\ell}_{\mathcal{B}}$, for a specifically chosen $\ell \in \mathcal{B}$ (we will later refer to this specific index as $\kappa_{\mathcal{B}}$ and we discuss its selection in Rule 5.84). Hence, we use lines 3–5 to construct this index set.

There is one last result that we need to establish before we present the our algorithm for partitioning S_{θ}. Given a basis $\mathcal{B}$ whose invariancy region is $(k-1)$-dimensional, we need to determine a strategy for selecting the index $\ell \in \mathcal{B}$ which we will use in a call to FindAdjacentKminus1($\mathcal{B}, \ell, \mathscr{B}$). For this purpose we introduce some additional notation. For each complementary basis $\mathcal{B}$ discovered during our computation, we keep track of a particular index in $\mathcal{B}$, which we denote $\kappa_{\mathcal{B}}$ and define according to the following rule:

Rule 5.84. (For defining $\kappa_{\mathcal{B}}$)

Suppose $\mathcal{B}$ was discovered by a pivot from basis $\mathcal{B}'$. Then we define $\kappa_{\mathcal{B}}$ so that:

1. $\kappa_{\mathcal{B}} = \overline{\imath}$ if there exists $i \in \mathcal{B}'$ such that $\mathcal{B} = (\mathcal{B}' \setminus \{i\}) \cup \{\overline{\imath}\}$ and $(T_{\mathcal{B}'}(\phi))_{i\overline{\imath}}$ is not identically zero.

2. $\kappa_{\mathcal{B}} = \overline{\jmath}$ if there exist distinct $i, j \in \mathcal{B}'$ such that $\mathcal{B} = (\mathcal{B}' \setminus \{i, j\}) \cup \{\overline{\imath}, \overline{\jmath}\}$ and $(T_{\mathcal{B}'}(\phi))_{i\overline{\imath}}$ is identically zero, and there exists $\phi' \in S_{\phi}$ such that $(T_{\mathcal{B}}(\phi'))_{j,\overline{\imath}} > 0$ and $(T_{\mathcal{B}}(\phi'))_{i,\overline{\jmath}} \neq 0$.

We note that, although we do not specifically include the recording of $\kappa_{\mathcal{B}}$ in Algorithm 5.3 or 5.4, we assume for the remainder of this work that: (i) for every basis discovered in Algorithm 5.3, $\kappa_{\mathcal{B}}$ is recorded whenever the "if statements" on lines 5 and 12 are satisfied, and (ii) for every basis discovered in Algorithm 5.4, $\kappa_{\mathcal{B}}$ is recorded whenever the "if statements" on lines 8 and 15 are satisfied. Now, consider the following proposition concerning the index $\kappa_{\mathcal{B}}$.

Proposition 5.85. Let distinct complementary bases $\mathcal{B}$ and $\mathcal{B}'$ be given such that: (i) $dim(\mathcal{IR}_{\mathcal{B}}) = k-1$, $|\mathcal{B} \cap \mathcal{B}'| \geq h-2$, and (iii) $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ are adjacent across $\hbar^{i}_{\mathcal{B}'}$ for some $i \in \mathcal{B}'$. Then we have $\mathcal{IR}_{\mathcal{B}} \subset \hbar^{\kappa_{\mathcal{B}}}_{\mathcal{B}}$.

Proof. Recognize that because $\mathcal{IR}_{\mathcal{B}}$ is $(k-1)$-dimensional and adjacent to $\mathcal{IR}_{\mathcal{B}'}$ across $\hbar^{i}_{\mathcal{B}'}$, we have $\mathcal{IR}_{\mathcal{B}} \subset \hbar^{i}_{\mathcal{B}'}$. We now consider the cases in which $\mathcal{B}$ was obtained by a diagonal pivot or an exchange pivot from $\mathcal{B}'$, beginning with the case of the diagonal pivot. If $\mathcal{B}$ was obtained from $\mathcal{B}'$ using a diagonal pivot as outlined in condition (1) of either Proposition 5.72 or 5.83 then there exists $i \in \mathcal{B}'$ such that $\mathcal{B} = (\mathcal{B}' \setminus \{i\}) \cup \{\bar{\imath}\}$ and $(T_{\hat{\mathcal{B}}'}(\phi))_{i,\bar{\imath}}$ is not identically zero. Performing the pivot on this element shows that the RHS of $T_{\mathcal{B}}(\phi)$ associated with $\bar{\imath}$ is equal to the RHS of $T_{\hat{\mathcal{B}}}(\phi)$ divided by $(T_{\hat{\mathcal{B}}}(\phi))_{i,\bar{\imath}}$. Hence, $\hbar^{i}_{\hat{\mathcal{B}}'} \subseteq \hbar^{\bar{\imath}}_{\mathcal{B}}$. Therefore, by letting $\kappa_{\mathcal{B}} = \bar{\imath}$ we have $\mathcal{IR}_{\mathcal{B}} \subset \hbar^{\kappa_{\mathcal{B}}}_{\mathcal{B}}$. Now suppose that $\mathcal{B}$ was obtained from $\mathcal{B}'$ using an exchange pivot as outlined in condition (2) of either Proposition 5.72 or 5.83. In this case there exist distinct $i, j \in \mathcal{B}' \setminus \{i\}$ such that $\mathcal{B} = (\mathcal{B}' \setminus \{i,j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$, $(T_{\mathcal{B}'}(\phi))_{i,\bar{\imath}}$ is identically zero and there exists $\phi' \in S_{\phi}$ such that $(T_{\mathcal{B}'}(\phi'))_{j,\bar{\imath}} > 0$ and $(T_{\mathcal{B}'}(\phi'))_{i,\bar{\jmath}} \neq 0$. We showed in the proof of Proposition 5.72 that in this case $\hbar^{i}_{\mathcal{B}'} \subseteq \hbar^{\bar{\jmath}}_{\mathcal{B}}$. Therefore, by letting $\kappa_{\mathcal{B}} = \bar{\jmath}$ we have $\mathcal{IR}_{\mathcal{B}} \subset \hbar^{\kappa_{\mathcal{B}}}_{\mathcal{B}}$. Hence, the claim of the proposition holds in both cases. □

We have now build the tools necessary for presenting the main result of this section, an strategy for paritioning S_{θ} given an initial full dimensional invariancy region. We give this strategy in Algorithm 5.5.

Each of the algorithms presented in this section has included the statement, "Assume that the sets $\mathrm{Z}_{\mathcal{B}}, \mathrm{E}_{\mathcal{B}}$ and $\mathrm{H}^{i}_{\mathcal{B}}$ for all $i \in \mathcal{B}$ have already been constructed." Line 9 of Algorithm 5.5 provides justification for this assumption, as these sets are constructed for each basis discovered during a call to either FindAdjacentFull or FindAdjacentKminus1. Also, when we say "Build $\mathrm{Z}_{\mathcal{B}}$," we assume that one does so by observing the RHS of $T_{\mathcal{B}}(\phi, \theta)$ and adding to $\mathrm{Z}_{\mathcal{B}}$ each index in $\mathcal{B}$ whose RHS element is identically zero.

Algorithm 5.5 PARTITION$S_\theta(\mathcal{B}_0)$ – Partition the parameter space S_θ, given an initial full dimensional invariancy region.
Input: An initial complementary basis $\mathcal{B}_0$ such that $dim(\mathcal{IR}_{\mathcal{B}_0}) = k$. (Assume that the sets $\mathrm{Z}_{\mathcal{B}_0}, \mathrm{E}_{\mathcal{B}_0}, \mathrm{H}^i_{\mathcal{B}_0}$ for all $i \in \mathcal{B}_0$ have already been constructed.)
Output: The partition of S_θ, denoted $\mathcal{P}$.

```
Let $\mathcal{S} = \{\mathcal{B}_0\}$ and $\mathscr{B} = \emptyset$.
while $\mathcal{S} \neq \emptyset$ do select $\mathcal{B}$ from $\mathcal{S}$.
    if $dim(\mathcal{IR}_{\mathcal{B}}) = k$ then run BUILDF($\mathcal{B}$) and then FINDADJACENTFULL($\mathcal{B}, \mathscr{B}$).
    else run FINDADJACENTKMINUS1($\mathcal{B}, \kappa_{\mathcal{B}}, \mathscr{B}$).
    Set $\mathcal{S} = \mathcal{S} \cup \overline{\mathrm{A}}_{\mathcal{B}}$.
    for $\mathcal{B}' \in \overline{\mathrm{A}}_{\mathcal{B}}$ do solve $NLP_D(\mathcal{B}')$ to obtain optimal solution $(\lambda^*, \phi^*, \sigma^*)$.
        if $\lambda^* > 0$ then label $\mathcal{IR}_{\mathcal{B}'}$ as full dimensional and add it to $\mathcal{P}$.
        else label $\mathcal{IR}_{\mathcal{B}'}$ as $(k-1)$-dimensional.
        Build $\mathrm{Z}_{\mathcal{B}'}$ and then run BUILDEANDH($\mathcal{B}'$).
Return $\mathcal{P}$.
```

We now return to our consideration of Example 5.6. The additional invariancy regions which are adjacent to $\mathcal{IR}_{\mathcal{B}_1^{5.6}}$ are those adjacent across $\hbar^{z_2}_{\mathcal{B}_1^{5.6}}$ and $\hbar^{w_3}_{\mathcal{B}_1^{5.6}}$. Recognize from Table 5.15 in Section 5.B that diagonal pivots on z_2 and w_3 are not possible. Furthermore, the only possible exchange pivot involving z_2 also involves w_1, and the only possible exchange pivot involving w_3 also involves w_1. NLP_A can be used to verify that both of these pivots yield adjacent regions. Hence, we obtain $\mathcal{B}_3^{5.6} = \{z_1, z_2, z_3, z_4\}$ and $\mathcal{B}_4^{5.6} = \{z_1, w_2, w_3, z_4\}$, whose associated tableaux are located in Tables 5.17 and 5.18 in Section 5.B. Before further considering either of these bases, we first consider $\mathcal{B}_2^{5.6}$. Recall that, although we claimed it to be the case, we have yet to verify that $dim(\mathcal{IR}_{\mathcal{B}_2^{5.6}}) \neq k$. For this purpose, we solve $NLP_D(\mathcal{B}_2^{5.6})$ which has an approximate optimal solution of $(\lambda^*, \phi^*, \sigma^*) = (0, -0.0603, -1.8889)$. This shows that, in fact, $dim(\mathcal{IR}_{\mathcal{B}_2^{5.6}}) \neq k$. Recall from our previous work and from Rule 5.84 that $\kappa_{\mathcal{B}_2^{5.6}} = z_2$. Thus, we now run FINDADJACENTKMINUS1($\mathcal{B}_2^{5.6}$,z_2,$\mathscr{B}$). In doing so we find that $GCD(B_2^{5.6}, z_2, w_1) = \frac{1}{15}$, $GCD(B_2^{5.6}, z_2, z_3) = \frac{1}{15}$, and $GCD(B_2^{5.6}, z_2, w_4) = 3\sigma + \frac{17}{3}$. As $GCD(B_2^{5.6}, z_2, w_4)$ is the only of these which is a nonconstant polynomial, we now solve $NLP_G(B_2^{5.6}, z_2, w_3)$ from which we obtain an approximate optimal solution of $(\lambda^{**}, \phi^{**}, \sigma^{**}) = (0, -0.503, -1.8889)$ which shows that $w_4 \in \mathrm{D}^{z_2}_{\mathcal{B}_2^{5.6}}$. We observe from $T_{\mathcal{B}_2^{5.6}}(\phi)$, found in Table 5.16 in Section 5.B, that the only possible pivot from $\mathcal{B}_2^{5.6}$ which involves w_4 is the exchange pivot which also involves w_1. Recognize, though, that this pivot will result in obtaining $\mathcal{B}_3^{5.6}$. We leave the remainder of the consideration of Example 5.6 to the reader, as the remainder of the steps for partitioning S_θ are analogous to steps we have already shown. Recall, though, that the solution to both examples can be observed in Tables 5.1 and 5.2. In the next section we discuss a technique for determining an initial basis with a full dimensional invariancy region.

5.5 Phase 1: Determining an initial feasible solution

In this section we develop a method for determining an initial feasible solution to the mpLCP (5.1) which provides a good starting point for our task of partitioning the parameter space S_θ. Thus, we seek a basis $\mathcal{B}_0$ such that $dim\,(\mathcal{IR}_{\mathcal{B}_0}) = k$. We present the algorithm for finding $\mathcal{B}_0$ at the end of this section.

We now discuss the techniques which we use to obtain an initial basis $\mathcal{B}_0$ such that $dim\,(\mathcal{IR}_{\mathcal{B}_0}) = k$. We assume throughout this discussion that $\mathbf{0} \in S_\theta$. Recognize that this assumption is not restrictive because it can be achieved by a simple translation when necessary. Define the augmented phase 1 multiparametric LCP, mpLCP$_{ph1}$:

$$\begin{aligned} w - M(\phi)z &= q(\phi,\sigma) + r\rho \\ w^\top z &= 0 \\ w, z &\geq 0 \end{aligned} \tag{5.104}$$

Here $\rho \in \mathbb{R}$ is an additional parameter and $r \in \mathbb{R}^h$ is defined so that, if we represent $q(\phi,\sigma)$ as $q + \triangle Q_{\bullet U}\phi + \triangle Q_{\bullet V}$, we have

$$r_i = \begin{cases} |q_i| + 1 & \text{if } q_i \leq 0 \\ 0 & \text{otherwise} \end{cases} \qquad \text{for each } i \in \{1,\ldots,h\}. \tag{5.105}$$

Notice that mpLPC$_{ph1}$ (5.104) is a variant of mpLCP (5.1) in which k is replaced by $k+1$. Therefore, all definitions and theory presented for mpLCP directly apply to mpLPC$_{ph1}$. Throughout the following discussion we will use a superscript $ph1$ to denote the phase 1 analogues of various sets and other notations we defined in Section 5.4. For example, the notation $\mathcal{IR}_{\mathcal{B}}^{ph1}$ and $(\mathscr{h}_{\mathcal{B}}^{i})^{ph1}$ represent the respective analogues of $\mathcal{IR}_{\mathcal{B}}$ and $\mathscr{h}_{\mathcal{B}}^{i}$ for mpLPC$_{ph1}$.

Recognize that we impose no lower or upper bound on the value of ρ. As a result, the phase 1 counterpart to any NLP presented for phase 2 can yield an unbounded solution. We note that this is not a problem since an unbounded maximization (minimization) problem still yields a strictly positive (negative) optimal value, which is the condition that must be verified for the majority of the NLPs we have introduced.

Proposition 5.86. The complementary basis

$$\mathcal{B}^* := \{1, \ldots, h\} \tag{5.106}$$

is feasible to mpLCP$_{ph1}$ (5.104) and $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$ is full dimensional.

Proof. Since $G(\phi)_{\bullet\mathcal{B}^*} = I$, $\mathcal{IR}^{ph1}_{\mathcal{B}^*} = \{(\phi, \sigma, \rho) \in S_\phi \times S_\sigma \times \mathbb{R} : q(\phi, \sigma) + r\rho \geq 0\}$. Observe from this and (5.105) that $(\phi, \sigma, \rho) = (\mathbf{0}, \mathbf{0}, 1) \in \mathcal{IR}^{ph1}_{\mathcal{B}^*}$. Thus $\mathcal{IR}^{ph1}_{\mathcal{B}^*} \neq \emptyset$ and therefore $\mathcal{B}^*$ is feasible to mpLCP$_{ph1}$. Furthermore, since the system of inequalities $q(\phi, \sigma) + r\rho \geq 0$ is satisfied strictly at $(\phi, \sigma, \rho) = (\mathbf{0}, \mathbf{0}, 1)$, there must exist $\epsilon > 0$ such that $B_\epsilon\left((\mathbf{0}, \mathbf{0}, 1)\right) \subset \mathcal{IR}^{ph1}_{\mathcal{B}^*}$. Hence, $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$ is full dimensional. □

Proposition 5.86 shows that a full dimensional invariancy region for mpLCP$_{ph1}$ is immediately available. Thus, a very simple strategy for determining an initial basis $\mathcal{B}_0$ is to determine the k-dimensional boundaries of $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$, determine the bases whose phase 1 invariancy regions are adjacent to $\mathcal{IR}^{ph1}_{\mathcal{B}^*}$ across each such boundary, and then repeat this procedure for each newly discovered invariancy region. Each time a new basis $\mathcal{B}$ is discovered, $NLP_D(\mathcal{B})$ (5.87) can be solved to determine whether or not $\mathcal{IR}_\mathcal{B}$ is full dimensional. We then continue partitioning $S_\theta \times \mathbb{R}$ in the same way that we discussed partitioning S_θ in Section 5.5, and stop once a basis with a full dimensional invariancy region is discovered. If no such basis is discovered throughout the procedure, we can conclude that no such basis exists. Note that if no basis exists which has a full dimensional invariancy region, then there is no need to search for bases whose invariancy regions are $(k-1)$-dimensional. Although this procedure is a brute force method, it serves as a good foundation for the procedure we will ultimately use.

Recognize that for any phase 1 invariancy region $\mathcal{IR}^{ph1}_{\mathcal{B}}$, the phase 2 invariancy region $\mathcal{IR}_\mathcal{B}$ is precisely the intersection of $\mathcal{IR}^{ph1}_{\mathcal{B}}$ with the hyperplane $\rho = 0$. Thus, in order to improve the technique discussed above, we would like to determine the k-dimensional boundaries of an invariancy region $\mathcal{IR}^{ph1}_{\mathcal{B}}$ across which we are most likely to find an adjacent invariancy region $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ such that the intersection of $\mathcal{IR}^{ph1}_{\mathcal{B}'}$ with the hyperplane $\rho = 0$ has dimension k. With this in mind, consider the following NLP and the subsequent related proposition.

$$\begin{aligned} NLP_S(\mathcal{B}) := \quad \min_{\phi, \sigma, \rho} \quad & \rho \\ \text{s.t.} \quad & g(\mathcal{B}) Adj(G(\phi)_{\bullet\mathcal{B}}) \left(q(\phi, \sigma) + r\rho\right) \geq 0 \\ & \phi \in S_\phi, \sigma \in S_\sigma \end{aligned} \tag{5.107}$$

Proposition 5.87. If $M(\phi)$ is a Q_0 matrix for all $\phi \in S_\phi$, then the mpLCP (5.1) is feasible if and only if there exists a complementary basis $\mathscr{B}$ for which $NLP_S(\mathscr{B})$ (5.107) has a nonpositive optimal value.

Proof. ($\Rightarrow$): If mpLCP (5.1) is feasible then there is a basis $\mathscr{B}'$ and some $\hat{\theta} = (\hat{\phi}, \hat{\sigma}) \in S_\theta$ such that $g(\mathscr{B}')Adj(G(\hat{\phi})_{\bullet\mathscr{B}'})q(\hat{\phi}, \hat{\sigma}) \geq 0$. Clearly in this case $(\phi, \sigma, \rho) = (\hat{\phi}, \hat{\sigma}, 0)$ is feasible to $NLP_S(\mathscr{B}')$ and thus the optimal value must be nonpositive.
($\Leftarrow$): Recall that $NLP_S(\mathscr{B}^*)$ has a feasible solution in which $\rho = 1$. Thus, since mpLCP (5.1) is equivalent to mpLCP^{ph1} with ρ fixed to 0, if there exists a basis $\mathscr{B}'$ such that $NLP_S(\mathscr{B}')$ is feasible for some $\hat{\rho} \leq 0$ then, since $\mathcal{K}(M(\phi))$ is convex for all ϕ such that $M(\phi)$ is Q_0, there must exist a basis $\mathscr{B}''$ such that $NLP_S(\mathscr{B}'')$ is feasible at $\rho = 0$. Therefore mpLCP must be feasible. □

Given a complementary basis $\mathscr{B}$, denote the optimal solution of $NLP_S(\mathscr{B})$ as $(\phi^*_{\mathscr{B}}, \sigma^*_{\mathscr{B}}, \rho^*_{\mathscr{B}})$ and define

$$EQ_{\mathscr{B}} := \{i \in \mathscr{B} : Adj(G(\phi^*_{\mathscr{B}})_{\bullet\mathscr{B}})\,(q(\phi^*_{\mathscr{B}}, \sigma^*_{\mathscr{B}}) + r\rho^*_{\mathscr{B}}) = 0\} \tag{5.108}$$

which is the set of indices in $\mathscr{B}$ whose corresponding defining constraints of $\mathcal{IR}^{ph1}_{\mathscr{B}}$ are binding at $(\phi^*_{\mathscr{B}}, \sigma^*_{\mathscr{B}}, \rho^*_{\mathscr{B}})$.

Proposition 5.88. Assume that $M(\phi)$ is a Q_0 matrix for all $\phi \in S_\phi$. Let a complementary basis $\mathscr{B}$ be given and let $(\phi^*_{\mathscr{B}}, \sigma^*_{\mathscr{B}}, \rho^*_{\mathscr{B}})$ represent the optimal solution of $NLP_S(\mathscr{B})$ (5.107). Suppose that there does not exist an $i \in EQ_{\mathscr{B}}$ such that a diagonal or exchange pivot can be made from $\mathscr{B}$ which involves index i. Then the following hold:

- If $\rho^*_{\mathscr{B}} > 0$, then mpLCP is infeasible.
- If $\rho^*_{\mathscr{B}} = 0$ and $dim\,(\mathcal{IR}_{\mathscr{B}}) < k$, then there does not exist a feasible complementary basis $\mathscr{B}'$ such that $dim\,(\mathcal{IR}_{\mathscr{B}'}) = k$.

Proof. If no diagonal or exchange pivots are possible which involve a particular index $i \in \mathscr{B}$, this indicates that the facet $cone\left(G(\phi)_{\bullet(\mathscr{B}\setminus\{i\})}\right)$ of the parametric complementary cone $\mathcal{C}(\phi, \mathscr{B})$ forms a boundary of $\mathcal{K}(M(\phi))$ for all $\phi \in S_\phi$. Thus, since $\mathcal{K}(M(\phi))$ is convex for each ϕ such that $M(\phi)$ is a Q_0 matrix, all phase 1 invariancy regions lie in the same semi-algebraic half-space defined by the hypersurface $(\mathscr{h}^i_{\mathscr{B}})^{ph1}$ that $\mathcal{IR}^{ph1}_{\mathscr{B}}$ lies in. Since this is true for all indices in $EQ_{\mathscr{B}}$, we have the following:

1. If the optimal value of $NLP_S(\mathcal{B})$ is strictly positive, no phase 1 invariancy region exists which intersects the hyperplane $\rho = 0$.
2. If the optimal value of $NLP_S(\mathcal{B})$ is zero, no phase 1 invariancy region other than $\mathcal{IR}_{\mathcal{B}}^{ph1}$ can have a nonempty intersection with the hyperplane $\rho = 0$.

Since $\mathcal{IR}_{\mathcal{B}}$ is the intersection of $\rho = 0$ and $\mathcal{IR}_{\mathcal{B}}^{ph1}$, the claim of the proposition follows. $\square$

Observe that Proposition 5.88 provides the following two simplifications of the brute force method: (i) it identifies a subset of the k-dimensional boundaries of an invariancy region which need to be checked for adjacent invariancy regions, and (ii) it provides a stopping criterion under which one may conclude that either the mpLCP (5.1) is infeasible or there do not exist any full dimensional invariancy regions for mpLCP. It now seems that we may be able to use the following strategy for obtaining an initial invariancy region of full dimension: Follow the same procedure outlined in Section 5.4 for partitioning S_θ, except, given any discovered complementary basis $\mathcal{B}$, only consider pivoting on the indices in $\mathcal{B} \cap EQ_{\mathcal{B}}$. There is, however, one flaw in this strategy that we will remedy shortly. To see this flaw, consider the following situation. Suppose we have a complementary basis $\mathcal{B}$ such that $\mathcal{IR}_{\mathcal{B}}^{ph1}$ is $(k+1)$-dimensional, but is contained in the half-space $\rho \geq 0$ and $dim(\mathcal{IR}_{\mathcal{B}}) < k$. Further suppose that we pivot on an index in $EQ_{\mathcal{B}}$ and obtain a new basis $\mathcal{B}'$ for which $dim(\mathcal{IR}_{\mathcal{B}'}^{ph1}) = k$ and the optimal value of $NLP_S(\mathcal{B}')$ is strictly negative. It is then possible to obtain a third basis $\mathcal{B}''$ by a pivot on an index in $EQ_{\mathcal{B}'}$ such that $\mathcal{IR}_{\mathcal{B}''}^{ph1}$ is $(k+1)$-dimensional, but is contained in the half-space $\rho \leq 0$ and $dim(\mathcal{IR}_{\mathcal{B}''}) < k$. See Figure 5.5 for a visual example of this situation with $k = 2$.

To ensure that this situation does not arise, when we encounter a phase 1 invariancy region $\mathcal{IR}_{\mathcal{B}}^{ph1}$ that is k-dimensional, we do something stronger than search for adjacent phase 1 invariancy regions. Instead we search for phase 1 invariancy regions which are not only adjacent to $\mathcal{IR}_{\mathcal{B}}^{ph1}$, but are also adjacent to a full dimensional region which we know $\mathcal{IR}_{\mathcal{B}}^{ph1}$ to be adjacent to. To ensure that this is possible, throughout phase 1, for every discovered invariancy region we record the following:

$$\mathscr{P}_{\mathcal{B}} := \text{ The basis for which we know } dim(\mathcal{IR}_{\mathscr{P}_{\mathcal{B}}}^{ph1}) = k+1 \text{ and } \mathcal{IR}_{\mathcal{B}}^{ph1} \text{ is adjacent to } \mathcal{IR}_{\mathscr{P}_{\mathcal{B}}}^{ph1}. \tag{5.109}$$

$$\iota_{\mathcal{B}} := \text{ The index in } \mathscr{P}_{\mathcal{B}} \text{ for which we know } \mathcal{IR}_{\mathcal{B}}^{ph1} \text{ is adjacent to } \mathcal{IR}_{\mathscr{P}_{\mathcal{B}}}^{ph1} \text{ along } (\mathscr{h}_{\mathscr{P}_{\mathcal{B}}}^{\iota_{\mathcal{B}}})^{ph1}. \tag{5.110}$$

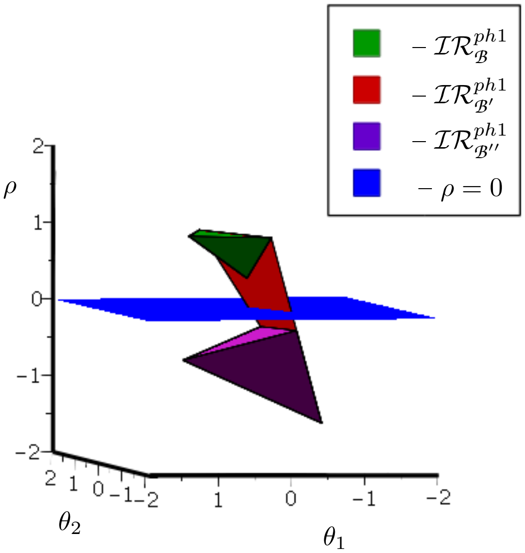


Figure 5.5: Example of pivots yielding $(k+1)$-dimensional phase 1 regions on either side of $\rho = 0$, but without generating a full dimensional phase 2 region.

Recognize that recording these values is always possible since we begin phase 1 by considering $\mathscr{B}^*$, which has a full dimensional phase 1 invariancy region. Throughout the rest of phase 1 we use the strategy outlined in Algorithm 5.6 to record $\mathscr{P}_{\mathscr{B}}$ and $\iota_{\mathscr{B}}$.

Algorithm 5.6 BuildPandι($\mathscr{B}$,$\mathscr{B}'$) – Construct $\mathscr{P}_{\mathscr{B}}$ and $\iota_{\mathscr{B}}$.
Input: A complementary basis $\mathscr{B}$ and the basis $\mathscr{B}'$ from which a diagonal or exchange pivot was performed in order to obtain $\mathscr{B}$.
Output: $\mathscr{P}_{\mathscr{B}'}$ and $\iota_{\mathscr{B}}$.

```
if $dim(\mathcal{IR}^{ph1}_{\mathscr{B}'}) = k+1$ then set $\mathscr{P}_{\mathscr{B}} = \mathscr{B}'$.
    if $\mathscr{B}'$ was obtained from $\mathscr{B}$ by a diagonal pivot involving an index $i \in \mathscr{B}$ satisfying condition (1) of Proposition 5.72 then set $\iota_{\mathscr{B}'} = i$.
    else we know $\mathscr{B}'$ was obtained from $\mathscr{B}$ by an exchange pivot involving a pair of indices $i, j \in \mathscr{B}$ satisfying condition (2) of Proposition 5.72. Thus, set $\iota_{\mathscr{B}'} = i$.
else set $\mathscr{P}_{\mathscr{B}} = \mathscr{P}_{\mathscr{B}'}$ and $\iota_{\mathscr{B}} = \iota_{\mathscr{B}'}$.
Return $\mathscr{P}_{\mathscr{B}}$ and $\iota_{\mathscr{B}}$.
```

We are almost ready to introduce our algorithm for obtaining an initial basis with full dimensional phase 2 invariancy region. Before we do so, however, we note that all of the NLPs, with the exception of NLP_{A2}, have analogous counterparts which are applicable for phase 1. For each NLP we denote the phase 1 analogue by adding a superscript of "ph1." Note that the only modification needed to obtain these NLPs is that in each $q(\phi, \sigma)$ be replaced by $q(\phi, \sigma) + r\rho$. Additionally, each of the procedures we outlined in algorithms in Section 5.4 also have counterparts which are applicable for phase 1, with the except of PartitionS_θ, of course. Many of these procedures require slight modifications, however, so we will soon present the phase 1 counterparts explicitly. Note that for each procedure, we denote the phase 1 counterpart by adding appending "_Ph1" to the name of the procedure. For example, BuildEandH_Ph1 denotes the phase 1 analogue of the phase 2 procedure BuildEandH. Before we present these procedures, first recall that, due to the result of Proposition 5.88, whenever the solution to NLP_S is nonnegative, we are not interested in finding all k-dimensional boundaries of a $(k+1)$-dimensional phase 1 invariancy region, but only the subset

$$\overline{\mathrm{F}}_{\mathcal{B}} := \left\{ i \in EQ_{\mathcal{B}} : (\hbar^i_{\mathcal{B}})^{ph1} \text{ forms a } k\text{-dimensional boundary of } \mathcal{IR}^{\phi}_{\mathcal{B}} \right\} \tag{5.111}$$

of k-dimensional boundaries whose associated indices are also in the set $EQ_{\mathcal{B}}$. Thus, there are two counterparts to the phase 2 procedure BuildF, which we denote BuildF_Ph1_1 and BuildF_Ph1_2. We use the former when the solution to NLP_S is nonnegative and thus return $\overline{\mathrm{F}}_{\mathcal{B}}$, otherwise we use the latter which returns $\mathrm{F}^{ph1}_{\mathcal{B}}$. We now present the phase 1 counterparts of each of the procedures we outlined in algorithms in Section 5.4.

Algorithm 5.7 BuildEandH_Ph1($\mathcal{B}$) – Build $\mathrm{E}^{ph1}_{\mathcal{B}}$ and $(\mathrm{H}^i_{\mathcal{B}})^{ph1}$ for each $i \in \mathcal{B}$.
Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}^{ph1}_{\mathcal{B}}) \geq k$. (Assume the set $\mathrm{Z}^{ph1}_{\mathcal{B}}$ has been constructed.)
Output: The sets $\mathrm{E}^{ph1}_{\mathcal{B}}$ and $(\mathrm{H}^i_{\mathcal{B}})^{ph1}$ for each $i \in \mathcal{B}$.

```
1: Let $\mathrm{E}^{ph1}_{\mathcal{B}} = \emptyset$.
2: Let $(\mathrm{H}^\ell_{\mathcal{B}})^{ph1} = \emptyset$ for each $\ell \in \mathcal{B}$.
3: for $i \in \left(\mathcal{B} \setminus \left(\mathrm{Z}^{ph1}_{\mathcal{B}} \cup \mathrm{E}^{ph1}_{\mathcal{B}}\right)\right)$ do
4:     for $j \in \left(\mathcal{B} \setminus \left(\mathrm{Z}^{ph1}_{\mathcal{B}} \cup \mathrm{E}^{ph1}_{\mathcal{B}} \cup \{i\}\right)\right)$ do
5:         if $j \notin (\mathrm{H}^i_{\mathcal{B}})^{ph1}$ then solve $NLP^{ph1}_H(\mathcal{B}, i, j)$ to obtain optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
6:             if $\lambda^* = 0$ then add $\left(j \cup (\mathrm{H}^j_{\mathcal{B}})^{ph1}\right)$ to $(\mathrm{H}^i_{\mathcal{B}})^{ph1}$.
7:             else if $\lambda^* < 0$ then add $i$ to $\mathrm{E}^{ph1}_{\mathcal{B}}$ and exit the for loop beginning on Line 4.
8: Return $\mathrm{E}^{ph1}_{\mathcal{B}}$ and $(\mathrm{H}^\ell_{\mathcal{B}})^{ph1}$ for each $\ell \in \mathcal{B}$.
```

Notice that because our procedure for obtaining phase 1 invariancy regions which are adjacent to phase 1 invariancy regions that are not full dimensional is not analogous to the procedure used in phase 2, we cannot yet provide an algorithm outlining the procedure FindAdjacentK_Ph1. We first present the following proposition, which serves as the phase 1 counterpart to Proposition 5.83.

Algorithm 5.8 BuildF_Ph1_1($\mathscr{B}$) – Build $\overline{\mathrm{F}}_{\mathscr{B}}$.

Input: A complementary basis $\mathscr{B}$ such that $dim(\mathcal{IR}_{\mathscr{B}}^{ph1}) = k+1$. (Assume that $NLP_S(\mathscr{B})$ has been solved and has a nonnegative optimal value. Also assume that the sets $EQ_{\mathscr{B}}$, $\mathrm{Z}_{\mathscr{B}}^{ph1}, \mathrm{E}_{\mathscr{B}}^{ph1}$ and $(\mathrm{H}_{\mathscr{B}}^{i})^{ph1}$ for all $i \in \mathscr{B}$ have been constructed.)

Output: The set $\mathrm{F}_{\mathscr{B}}$.

```
Let $\overline{\mathrm{F}}_{\mathscr{B}} = \emptyset$.
for $i \in \left(EQ_{\mathscr{B}} \setminus \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{E}_{\mathscr{B}} \cup \overline{\mathrm{F}}_{\mathscr{B}}\right)\right)$ do solve $NLP_F^{ph1}(\mathscr{B}, i)$ to find optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
    if $\lambda^* > 0$ then add $\left(i \cup (\mathrm{H}_{\mathscr{B}}^{i})^{ph1}\right)$ to $\overline{\mathrm{F}}_{\mathscr{B}}$.
Return $\overline{\mathrm{F}}_{\mathscr{B}}$.
```

Algorithm 5.9 BuildF_Ph1_2($\mathscr{B}$) – Build $\overline{\mathrm{F}}_{\mathscr{B}}$.

Input: A complementary basis $\mathscr{B}$ such that $dim(\mathcal{IR}_{\mathscr{B}}^{ph1}) = k+1$. (Assume that $NLP_S(\mathscr{B})$ has been solved and has a strictly negative optimal value. Also assume that the sets $EQ_{\mathscr{B}}$, $\mathrm{Z}_{\mathscr{B}}^{ph1}, \mathrm{E}_{\mathscr{B}}^{ph1}$ and $(\mathrm{H}_{\mathscr{B}}^{i})^{ph1}$ for all $i \in \mathscr{B}$ have been constructed.)

Output: The set $\mathrm{F}_{\mathscr{B}}$.

```
Let $\mathrm{F}_{\mathscr{B}}^{ph1} = \emptyset$.
for $i \in \left(\mathscr{B} \setminus \left(\mathrm{Z}_{\mathscr{B}} \cup \mathrm{E}_{\mathscr{B}} \cup \mathrm{F}_{\mathscr{B}}^{ph1}\right)\right)$ do solve $NLP_F^{ph1}(\mathscr{B}, i)$ to find optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
    if $\lambda^* > 0$ then add $\left(i \cup (\mathrm{H}_{\mathscr{B}}^{i})^{ph1}\right)$ to $\mathrm{F}_{\mathscr{B}}^{ph1}$.
Return $\mathrm{F}_{\mathscr{B}}^{ph1}$.
```

Proposition 5.89. Let distinct bases $\mathscr{B}$ and $\mathscr{B}'$ be given for which $dim(\mathcal{IR}_{\mathscr{B}}^{ph1}) = k$ and $|\mathscr{B} \cap \mathscr{B}'| \geq h-2$. Furthermore, let $\ell \in \mathscr{B}$ be the index for which we know $\mathcal{IR}_{\mathscr{B}}^{ph1} \subset (\hslash_{\mathscr{B}}^{\ell})^{ph1}$. Then $\mathcal{IR}_{\mathscr{B}'}^{ph1}$ is adjacent to $\mathcal{IR}_{\mathscr{P}_{\mathscr{B}}}^{ph1}$ along $(\hslash_{\mathscr{B}}^{\ell})^{ph1}$ if and only if there exists $i \in (\mathrm{D}_{\mathscr{B}}^{\ell})^{ph1}$ such that one of the following conditions holds:

1. $\mathscr{B}' = (\mathscr{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ and $(T_{\mathscr{B}}^{ph1}(\phi))_{i,\bar{\imath}}$ is not identically zero.
2. There exists an additional index $j \in \mathscr{B} \setminus \{i\}$ such that $\mathscr{B}' = (\mathscr{B} \setminus \{i,j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$, $(T_{\mathscr{B}}^{ph1}(\phi))_{i,\bar{\imath}}$ is identically zero, there exists $\phi' \in S_\phi$ such that $\left(T_{\mathscr{B}}^{ph1}(\phi')\right)_{j,\bar{\imath}} > 0$ and $\left(T_{\mathscr{B}}^{ph1}(\phi')\right)_{i,\bar{\jmath}} \neq 0$, and the following NLP has a strictly positive optimal value:

$$
\begin{array}{lll}
NLP_{A2}^{ph1}(\mathscr{B}, i, j) := & & \\
\quad \max\limits_{\lambda,\phi,\sigma} & \lambda & \\
\quad \text{s.t.} & g(\mathscr{P}_{\mathscr{B}}) \left(Adj(G(\phi)_{\bullet\mathscr{P}_{\mathscr{B}}})\right)_{\xi\bullet} (q(\phi,\sigma) + r\rho) \geq \lambda \mathbf{1} & \forall\, \xi \in \left(\mathscr{P}_{\mathscr{B}} \setminus \left(\mathrm{Z}_{\mathscr{P}_{\mathscr{B}}}^{ph1} \cup (\mathrm{H}_{\mathscr{P}_{\mathscr{B}}}^{\iota_{\mathscr{B}}})^{ph1} \cup \{\iota_{\mathscr{B}}\}\right)\right) \\
 & \left(Adj(G(\phi)_{\bullet\mathscr{P}_{\mathscr{B}}})\right)_{\iota_{\mathscr{B}}\bullet} (q(\phi,\sigma) + r\rho) = 0 & \\
 & g(\mathscr{B}') \left(Adj(G(\phi)_{\bullet\mathscr{B}'})\right)_{\xi\bullet} (q(\phi,\sigma) + r\rho) \geq \lambda \mathbf{1} & \forall\, \xi \in \left(\mathscr{B}' \setminus \left(\mathrm{Z}_{\mathscr{B}'}^{ph1} \cup (\mathrm{H}_{\mathscr{B}'}^{\bar{\jmath}})^{ph1} \cup \{\bar{\jmath}\}\right)\right) \\
 & \phi \in S_\phi, \sigma \in S_\sigma &
\end{array}
\tag{5.112}
$$

Algorithm 5.10 FindAdjacentFull_Ph1($\mathcal{B}$,$\mathscr{B}$) – Determine all previously undiscovered invariancy regions which are adjacent to a given full dimensional phase 1 invariancy region.

Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}^{ph1}_{\mathcal{B}}) = k+1$ and the set $\mathscr{B}$ of previously discovered bases. (Assume that $NLP_S(\mathcal{B})$ has been solved and that the sets $\mathrm{Z}^{ph1}_{\mathcal{B}}, \mathrm{E}^{ph1}_{\mathcal{B}}, (\mathrm{H}^{i}_{\mathcal{B}})^{ph1}$ for all $i \in \mathcal{B}$, and either $\overline{\mathrm{F}}_{\mathcal{B}}$ or $\mathrm{F}^{ph1}_{\mathcal{B}}$ have been constructed.).

Output: The set $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and an updated version of $\mathscr{B}$.

```
Let $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}} = \emptyset$.
Select an arbitrary $\phi' \in S_{\phi}$.
Let $(\hat{\rho}, \hat{\phi}, \hat{\sigma})$ denote the optimal solution of $NLP_S(\mathcal{B})$.
if $\hat{\rho} < 0$ then let $\mathscr{F}$ denote $\mathrm{F}^{ph1}_{\mathcal{B}}$.
else let $\mathscr{F}$ denote $\overline{\mathrm{F}}_{\mathcal{B}}$.
for $i \in \mathscr{F}$ do
    Let $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$.
    if $\mathcal{B}' \notin \mathscr{B}$ and $(T^{ph1}_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero then add $\mathcal{B}'$ to $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and $\mathscr{B}$ and run BuildPandι($\mathcal{B}$,$\mathcal{B}'$).
    else
        for $j \in \mathcal{B} \setminus \{i\}$ do
            Let $\mathcal{B}'' = (\mathcal{B} \setminus \{i,j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$.
            if $\mathcal{B}'' \notin \mathscr{B}$, $(T^{ph1}_{\mathcal{B}}(\phi'))_{j,\bar{\imath}} > 0$ and $(T^{ph1}_{\mathcal{B}}(\phi'))_{i,\bar{\jmath}} \neq 0$ then
                Perform pivots to find $T^{ph1}_{\mathcal{B}''}(\phi)$, then build $\mathrm{Z}^{ph1}_{\mathcal{B}''}$ and run BuildEandH_Ph1($\mathcal{B}''$).
                Solve $NLP^{ph1}_{A}(\mathcal{B}, i, j)$ to obtain optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
                if $\lambda^* > 0$ then add $\mathcal{B}''$ to $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and $\mathscr{B}$ and run BuildPandι($\mathcal{B}$,$\mathcal{B}'$).
Return $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and $\mathscr{B}$.
```

Proof. It is clear from analogous arguments to those given in Section 5.4, namely Proposition 5.83, together with Observation 5.80, (5.99), (5.100) and (5.101), that $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}} \cap (\hbar^{\ell}_{\mathcal{B}})^{ph1} \cap (\hbar^{i}_{\mathcal{B}})^{ph1}\right) = k$ if and only if $i \in (\mathrm{D}^{\ell}_{\mathcal{B}})^{ph1}$. Furthermore, it is also clear that if $\mathcal{B}$ was obtained from $\mathscr{P}_{\mathcal{B}}$ by a single diagonal as prescribed in condition (1) or a single exchange pivot as prescribed in condition (2) then $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}} \cap (\hbar^{\ell}_{\mathcal{B}})^{ph1} \cap (\hbar^{\iota_{\mathcal{B}}}_{\mathscr{P}_{\mathcal{B}}})^{ph1}\right) = k$. We also have $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}} \cap (\hbar^{\ell}_{\mathcal{B}})^{ph1} \cap (\hbar^{i}_{\mathcal{B}})^{ph1} \cap (\hbar^{\iota_{\mathcal{B}}}_{\mathscr{P}_{\mathcal{B}}})^{ph1}\right) = k$. Now suppose that $\mathcal{B}$ was obtained from $\mathscr{P}_{\mathcal{B}}$ by a sequence of pivots to intermediate bases, with each pivot satisfying either condition (1) or (2). In the case of pivots satisfying condition (1), the adjacency of each newly discovered region to $\mathcal{IR}_{\mathscr{P}_{\mathcal{B}}}$ along $(\hbar^{\ell}_{\mathcal{B}})^{ph1}$ is clear due to the uniqueness of shared k-dimesional boundaries for regions obtained from diagonal pivots. In the case of pivots satisfying condition (2), recognize from (5.112) that the first set of constraints of NLP^{ph1}_{A2} ensure that at any solution for which $\lambda > 0$, all defining constraints of $\mathcal{IR}^{ph1}_{\mathscr{P}_{\mathcal{B}}}$ are satisfied strictly except for those implied by $(\hbar^{\ell}_{\mathcal{B}})^{ph1}$. The second constraint ensures that solutions lie on $(\hbar^{\ell}_{\mathcal{B}})^{ph1}$. Finally, the third set of constraints ensure that at any solution for which $\lambda > 0$, all defining constraints of the newly discovered invariancy region are satisfied strictly except for those implied by $(\hbar^{\ell}_{\mathcal{B}})^{ph1}$.

Recognize that this can only hold if $dim\left(\mathcal{IR}^{ph1}_{\mathcal{B}} \cap (\hbar^{\ell}_{\mathcal{B}})^{ph1} \cap (\hbar^{i}_{\mathcal{B}})^{ph1} \cap (\hbar^{\iota_{\mathcal{B}}}_{\mathscr{P}_{\mathcal{B}}})^{ph1}\right) = k$. Hence, we can conclude that $\mathcal{IR}^{ph1}_{\mathscr{P}_{\mathcal{B}}}$ and $\mathcal{IR}^{ph1}_{\mathcal{B}}$ are adjacent along $(\hbar^{\ell}_{\mathcal{B}})^{ph1}$ regardless of the number of pivots taken to obtain $\mathcal{B}$ from $\mathscr{P}_{\mathcal{B}}$, so long as each pivot satisfies either condition (1) or (2). The remainder of the proof is analogous to those of Propositions 5.72 and 5.83. □

We now present FindAdjacentK_Ph1 which is modified from FindAdjacentKminus1 in order to account for the differences we mentioned earlier. This procedure is presented in Algorithm 5.11.

Algorithm 5.11 FindAdjacentK_Ph1($\mathcal{B}$,ℓ,$\mathscr{B}$) – Determine all previously undiscovered invariancy regions which are adjacent to a given k-dimensional phase 1 invariancy region, and furthermore, are also adjacent to a specific $(k+1)$-dimensional phase 1 invariancy region.
Input: A complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}^{ph1}_{\mathcal{B}}) = k$, an index $\ell \in \mathcal{B}$ for which we know $\mathcal{IR}^{ph1}_{\mathcal{B}} \subset (\hbar^{\ell}_{\mathcal{B}})^{ph1}$ and the set $\mathscr{B}$ of previously discovered bases. (Assume that the sets $\mathrm{Z}^{ph1}_{\mathcal{B}}$,$\mathrm{E}^{ph1}_{\mathcal{B}}$, and $(\mathrm{H}^{i}_{\mathcal{B}})^{ph1}$ for all $i \in \mathcal{B}$ have already been constructed.)
Output: The set $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and an updated version of $\mathscr{B}$.

```
Let $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}} = \emptyset$ and $(\mathrm{D}^{\ell}_{\mathcal{B}})^{ph1} = \emptyset$.
Select an arbitrary $\phi' \in S_{\phi}$.
**for** $j \in \mathcal{B} \setminus \{\ell\}$ **do** compute $GCD^{ph1}(\mathcal{B}, \ell, j)$.
    **if** $GCD^{ph1}(\mathcal{B}, \ell, j)$ is a nonconstant polynomial **then** solve $NLP^{ph1}_{G}(\mathcal{B}, \ell, j)$ to obtain the optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
        **if** $\lambda^* = 0$ **then** add $j$ to $(\mathrm{D}^{\ell}_{\mathcal{B}})^{ph1}$.
**for** $i \in (\mathrm{D}^{\ell}_{\mathcal{B}})^{ph1}$ **do**
    Let $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$
    **if** $\mathcal{B}' \notin \mathscr{B}$ and $(T^{ph1}_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero **then** add $\mathcal{B}'$ to $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and $\mathscr{B}$ and run BuildPAndι($\mathcal{B}, \mathcal{B}'$).
    **else**
        **for** $j \in \mathcal{B} \setminus \{i\}$ **do**
            Let $\mathcal{B}'' = (\mathcal{B} \setminus \{i, j\}) \cup \{\bar{\imath}, \bar{\jmath}\}$
            **if** $\mathcal{B}'' \notin \mathscr{B}$, $(T^{ph1}_{\mathcal{B}}(\phi'))_{j,\bar{\imath}} > 0$ and $(T^{ph1}_{\mathcal{B}}(\phi'))_{i,\bar{\jmath}} \neq 0$ **then**
                Compute $T^{ph1}_{\mathcal{B}''}(\phi)$, then build $\mathrm{Z}^{ph1}_{\mathcal{B}''}$.
                Run BuildEandH_Ph1($\mathcal{B}''$) and BuildPAndι($\mathcal{B}, \mathcal{B}''$).
                Solve $NLP^{ph1}_{A2}(\mathcal{B}, i, j)$ to obtain optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
                **if** $\lambda^* > 0$ **then** add $\mathcal{B}''$ to $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and $\mathscr{B}$.
Return $\overline{\mathrm{A}}^{ph1}_{\mathcal{B}}$ and $\mathscr{B}$.
```

We are now able to present the algorithm which we use to obtain an initial complementary basis whose phase 2 invariancy region is k-dimensional. We do so in Algorithm 5.12. Afterwards we present a proposition in which we prove the correctness of the proposed method.

Algorithm 5.12 FindInitialBasis – Find an initial complementary basis having a full dimensional phase 2 invariancy region.
Input: None, other than an instance of mpLCP.
Output: An initial basis $\mathcal{B}_0$ such that $dim(\mathcal{IR}_{\mathcal{B}_0}) = k$ and the set $\mathscr{B}$.

```
1: Let $\mathcal{S} = \{\mathcal{B}^*\}$ and $\mathscr{B} = \emptyset$.
2: while $\mathcal{S} \neq \emptyset$ do select $\mathcal{B}$ from $\mathcal{S}$.
3:     Solve $NLP_S(\mathcal{B})$ to obtain optimal solution $(\rho', \phi', \sigma')$.
4:     if $\rho' < 0$ then
5:         if $dim(\mathcal{IR}_{\mathcal{B}}^{ph1}) = k + 1$ then solve $NLP_D(\mathcal{B})$ to find optimal solution $(\lambda'', \phi'', \sigma'')$.
6:             if $\lambda'' > 0$ then STOP. Set $\mathcal{B}_0 = \mathcal{B}$ and return $\mathcal{B}_0$ and $\mathscr{B}$.
7:             else run BuildF_Ph1_2($\mathcal{B}$) and then FindAdjacentFull_Ph1($\mathcal{B}, \mathscr{B}$).
8:         else run FindAdjacentK_Ph1($\mathcal{B}, \kappa_{\mathcal{B}}, \mathscr{B}$).
9:     else
10:        if $dim(\mathcal{IR}_{\mathcal{B}}^{ph1}) = k + 1$ then run BuildF_Ph1_1($\mathcal{B}$) and then FindAdjacentFull_Ph1($\mathcal{B}, \mathscr{B}$).
11:            if $\overline{\mathrm{A}}_{\mathcal{B}}^{ph1} = \emptyset$ and $\rho' \geq 0$ then STOP. There is no $\mathcal{B}'$ such that $dim(\mathcal{IR}_{\mathcal{B}'}) = k$.
12:        else run FindAdjacentK_Ph1($\mathcal{B}, \kappa_{\mathcal{B}}, \mathscr{B}$).
13:    Set $\mathcal{S} = \mathcal{S} \cup \overline{\mathrm{A}}_{\mathcal{B}}^{ph1}$.
14:    for $\mathcal{B}' \in \overline{\mathrm{A}}_{\mathcal{B}}^{ph1}$ do solve $NLP_D^{ph1}(\mathcal{B}')$ to find optimal solution $(\lambda^*, \rho^*, \phi^*, \sigma^*)$.
15:        if $\lambda^* > 0$ then label $\mathcal{IR}_{\mathcal{B}'}^{ph1}$ as full dimensional.
16:        else label $\mathcal{IR}_{\mathcal{B}'}^{ph1}$ as $k$-dimensional.
17:        Build $\mathrm{Z}_{\mathcal{B}'}^{ph1}$ and then run BuildEandH_Ph1($\mathcal{B}'$).
```

Proposition 5.90. Given an instance of mpLCP as described in (5.1), if $M(\phi)$ is sufficient for all ϕ in S_ϕ and there exists a complementary basis $\mathcal{B}$ such that $dim(\mathcal{IR}_{\mathcal{B}}) = k$, Algorithm 5.12 will return a complementary basis $\mathcal{B}'$ such that $dim(\mathcal{IR}_{\mathcal{B}'}) = k$.

Proof. Recognize that Algorithm 5.12 only ceases if a (i) STOP command is reached on either line 6 or 11, or (ii) there are no more bases in $\mathcal{S}$ to explore. Clearly if the STOP command on line 6 is reached the claim of the proposition holds since a full dimensional phase 2 invariancy region is returned. On the other hand, if the STOP command on line 11 is reached, the claim of the proposition is also satisfied since the correctness of this stopping criterion is proved in Proposition 5.88. Now consider the case in which there are no more bases in $\mathcal{S}$ to explore. The theory we have developed in this section ensures that, given a complementary basis $\mathcal{B}$ discovered during phase 1 for which $dim(\mathcal{IR}_{\mathcal{B}}) \neq k$, the only indices in $\mathcal{B}$ for which we do not consider performing pivots are precisely those which have no possibility of yielding a new basis $\mathcal{B}'$ such that $dim(\mathcal{IR}_{\mathcal{B}'}) = k$. Hence, the result of the proposition holds in this case as well. □

We return to Example 5.5 and briefly discuss how the theory presented here can be used to obtain an initial basis. We omit the consideration of phase 1 for Example 5.6 because only one iteration is necessary. It is easy to verify that the optimal value of $NLP_S(\mathscr{B}^{*\,5.6})$ (5.107) is nonpositive and the optimal value of $NLP_D(\mathscr{B}^{*\,5.6})$ (5.87) is strictly positive, and so $\mathscr{B}^{*\,5.6}$ serves as the initial basis for Example 5.6. Now, from Table 5.4 in Section 5.A we construct $NLP_S(\mathscr{B}^{*\,5.5})$:

$$
\begin{array}{ll}
\min\limits_{\phi,\rho} & \rho \\
\text{s.t.} & 3 \geq 0 \\
 & -2 - \phi_1 + 3\rho \geq 0 \\
 & 0 \geq 0 \\
 & 0 \geq 0 \\
 & 0 \geq 0 \\
 & \phi_1 + \phi_2 \leq 1 \\
 & \phi_1, \phi_2 \geq 0
\end{array}
$$

The approximate optimal solution is $(\phi_1^*, \phi_2^*, \rho^*) = (0, 0, 0.6667)$. This shows that $\mathcal{IR}^{ph1}_{\mathscr{B}^{*\,5.5}}$ does not intersect the hyperplane $\rho = 0$ and thus $\mathscr{B}^{*\,5.5}$ is infeasible for mpLCP. From (5.108) we find that $EQ_{\mathscr{B}^{*\,5.5}} = \{w_2, w_3, w_4, w_5\}$. However, recognize that $\mathrm{Z}^{ph1}_{\mathscr{B}} = \{w_3, w_4, w_5\}$. Hence, w_2 is the only variable in $\mathcal{IR}^{ph1}_{\mathscr{B}^{*\,5.5}}$ which is considered during the call to BuildF_Ph1. The optimal value of $NLP^{ph1}_F(\mathscr{B}^{*\,5.5})$ (see (5.88) for NLP_F) is unbounded, which shows that $(\hbar^{w_2}_{\mathscr{B}^{*\,5.5}})^{ph1}$ forms a k-dimensional boundary of $\mathcal{IR}^{ph1}_{\mathscr{B}^{*\,5.5}}$. Now we consider pivots on w_2. It can be observed from $T^{ph1}_{\mathscr{B}^{*\,5.5}}(\phi)$, found in Table 5.4 in Section 5.A, that although a diagonal pivot from w_2 is not possible, exchange pivots involving w_2 along with w_3, w_4 or w_5 are possible. The respective approximate optimal solutions of $NLP^{ph1}_A(\mathscr{B}^{*\,5.5}, w_2, w_3)$, $NLP^{ph1}_A(\mathscr{B}^{*\,5.5}, w_2, w_4)$, and $NLP^{ph1}_A(\mathscr{B}^{*\,5.5}, w_2, w_5)$ (see (5.94) for NLP_A), given in the order $(\lambda, \rho, \phi_1, \phi_2)$, are $(3, 0.7112, 0.1336, 0.1336)$, $(3, 0.7459, 0.2377, 0)$ and $(3, 0.7567, 0.2701, 0)$. This shows that the bases $\mathscr{B}^{5.5}_i = \{w_1, z_2, z_3, w_4, w_5\}$, $\mathscr{B}^{5.5}_{ii} = \{w_1, z_2, w_3, z_4, w_5\}$, and $\mathscr{B}^{5.5}_{iii} = \{w_1, z_2, w_3, w_4, z_5\}$ all yield phase 1 invariancy regions adjacent to $\mathcal{IR}^{ph1}_{\mathscr{B}^{*\,5.5}}$ across $(\hbar^{w_2}_{\mathscr{B}^{*\,5.5}})^{ph1}$. Hence, we set $\mathscr{P}_{\mathscr{B}^{5.5}_i} = \mathscr{P}_{\mathscr{B}^{5.5}_{ii}} = \mathscr{P}_{\mathscr{B}^{5.5}_{iii}} = \mathscr{B}^{*\,5.5}$ and $\iota_{\mathscr{B}^{5.5}_i} = \iota_{\mathscr{B}^{5.5}_{ii}} = \iota_{\mathscr{B}^{5.5}_{iii}} = w_2$. Note that the tableaux associated with $\mathscr{B}^{5.5}_i$, $\mathscr{B}^{5.5}_{ii}$ and $\mathscr{B}^{5.5}_{iii}$ can be found in respective order in Tables 5.5–5.7 in Section 5.A. We next consider basis $\mathscr{B}^{5.5}_i$. The approximate optimal solution of $NLP^{ph1}_D(\mathscr{B}^{5.5}_i)$ (see (5.87) for NLP_D) is $(0, 0, 0, 0.6667)$, which shows that $\mathcal{IR}^{ph1}_{\mathscr{B}^{5.5}_i}$ is not full dimensional. Thus, we run FindAdjacentK_Ph1. Recognize that $\kappa_{\mathscr{B}^{5.5}_i} = z_3$ and $\mathscr{B} = \{\mathscr{B}^{*\,5.5}, \mathscr{B}^{5.5}_i, \mathscr{B}^{5.5}_{ii}, \mathscr{B}^{5.5}_{iii}\}$. In doing so we find $GCD^{ph1}(\mathscr{B}^{5.5}_i, z_3, w_1) = \frac{1}{2}$, $GCD^{ph1}(\mathscr{B}^{5.5}_i, z_3, z_2) = \frac{1}{4}\phi_1 - \frac{3}{4}\rho + \frac{1}{2}$, and $GCD^{ph1}(\mathscr{B}^{5.5}_i, z_3, w_4) = GCD^{ph1}(\mathscr{B}^{5.5}_i, z_3, w_5) = \frac{1}{2}\phi_1 - \frac{3}{2}\rho + 1$. We also find that the optimal values of $NLP^{ph1}_G(\mathscr{B}^{5.5}_i, z_3, z_2)$,

$NLP_G^{ph1}(\mathcal{B}_i^{5.5}, z_3, w_4)$ and $NLP_G^{ph1}(\mathcal{B}_i^{5.5}, z_3, w_5)$ (see (5.102) for NLP_G) are all zero, so we have $(\mathrm{D}^{z_3}_{\mathcal{B}_i^{5.5}})^{ph1} = \{z_2, w_4, w_5\}$. We observe from Table 5.5 in Section 5.A that for each variable in $(\mathrm{D}^{z_3}_{\mathcal{B}_i^{5.5}})^{ph1}$ a diagonal pivot is possible. Performing these diagonal pivots result in new bases $\mathcal{B}_{iv}^{5.5} = \{w_1, w_2, z_3, w_4, w_5\}$, $\mathcal{B}_v^{5.5} = \{w_1, z_2, z_3, z_4, w_5\}$, and $\mathcal{B}_{vi}^{5.5} = \{w_1, z_2, z_3, w_4, z_5\}$. The tableaux associated with these bases can be found in Tables 5.8–5.10 in Section 5.A. We note that both $\mathcal{IR}_{\mathcal{B}_v^{5.5}}$ and $\mathcal{IR}_{\mathcal{B}_{vi}^{5.5}}$ are k-dimensional, while $\mathcal{IR}_{\mathcal{B}_{iv}^{5.5}}$ is full dimensional. However, $\mathcal{IR}_{\mathcal{B}_{iv}^{5.5}}$ also has another very interesting property, because $\mathcal{IR}_{\mathcal{B}_{iv}^{5.5}} = \mathcal{IR}_{\mathcal{B}^{*\,5.5}}$. We discuss this phenomenon in more detail in Section 5.6. At this point we claim that the remainder of the work needed to complete phase 1 for Example 5.5 is analogous to work we have already shown. In fact, a single diagonal pivot from either $\mathcal{B}_v^{5.5}$ or $\mathcal{B}_{vi}^{5.5}$ will reveal $\mathcal{B}_0^{5.5} = \{w_1, z_2, z_3, z_4, z_5\}$.

5.6 A note on obtaining non-overlapping invariancy regions

It is important to recognize that a partition of S_θ for mpLCP is not unique, in general. Hence, if one is not careful, when attempting to partition S_θ it is possible to generate invariancy regions $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$, associated with distinct complementary bases $\mathcal{B}$ and $\mathcal{B}'$, such that $dim(\mathcal{IR}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}) = k$. In fact, this very situation arose during our phase 1 examination of Example 5.5. In this section we pose two important research questions, which we present directly after the following two definitions.

Definition 5.91. Given distinct complementary bases $\mathcal{B}$ and $\mathcal{B}'$, we say that $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ *overlap* if $dim(\mathcal{IR}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}) = k$. Otherwise, they are *non-overlapping.*

Definition 5.92. Given an feasible instance of mpLCP, let $\mathcal{P}$ be the output of Algorithm 5.5. We say that $\mathcal{P}$ is a *true partition* of S_θ if no two invariancy regions in $\mathcal{P}$ overlap. Otherwise, we say that $\mathcal{P}$ is a *quasi-partition.*

Now consider these two important research questions.

Question 5.93. Given a feasible instance of mpLCP, what conditions are needed to ensure that a true partition of S_θ exists?

Question 5.94. When a true partition of S_θ exists, what precautions can be taken to ensure that one is obtained?

The remainder of this section is devoted to the consideration of these questions. We now introduce propositions whose results will allow us to establish an answer to Question 5.93.

Lemma 5.95. Given distinct complementary bases $\mathcal{B}$ and $\mathcal{B}'$, the invariancy regions $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ overlap if and only if there exists $\Phi \subseteq S_\phi$ such that $dim(\Phi) = p$ and $dim(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}')) = k - p$ for all $\phi \in \Phi$.

Proof. Recognize that the set $\mathcal{IR}_{\mathcal{B}} \cap \mathcal{IR}_{\mathcal{B}'}$ satisfies Property 5.38. Thus, by observing (5.22), (5.29) and Proposition 5.36 we conclude that the result of this lemma follows directly from Proposition 5.49. □

Proposition 5.96. Let distinct complementary bases $\mathcal{B}$ and $\mathcal{B}'$ be given such that the invariancy regions $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ overlap. Then there exist index sets $J \subset \mathcal{B}$ and $J' \subset \mathcal{B}'$ such that the following hold:

1. $k - p \leq |J| \leq k - p + 1$ and $k - p \leq |J'| \leq k - p + 1$.

2. $dim(cone(G(\phi)_{\bullet J}) \cap cone(G(\phi)_{\bullet J'})) \geq k - p$.

3. $aff(\mathcal{AS}(\phi)) \subseteq span(G(\phi)_{\bullet J^*})$ for all $\phi \in S_\phi$, where $aff(\mathcal{AS}(\phi))$ represents the affine hull of $\mathcal{AS}(\phi)$ and $J^* = \begin{cases} J & \text{if } |J| \geq |J'| \\ J' & \text{otherwise} \end{cases}$.

Proof. From Lemma 5.95 we have that there exists $\Phi \subseteq S_\phi$ such that $dim(\Phi) = p$ and $dim(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}')) = k - p$ for all $\phi \in \Phi$. This shows that for each $\phi \in \Phi$ the intersections $\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})$ and $\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}')$ must be contained within the boundaries of the respective parametric complementary cones. Furthermore, since for all $\phi \in S_\phi$ we have $dim(\mathcal{AS}(\phi)) = k-p$, for each $\phi \in S_\phi$ and each $\hat{\mathcal{B}} \in \{\mathcal{B}, \mathcal{B}'\}$ the intersection $\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \hat{\mathcal{B}})$ must occur either within a $(k-p)$-dimensional facet of $\mathcal{C}(\phi, \hat{\mathcal{B}})$, or within the convex hull of two $(k-p)$-dimensional facets (and therefore within a $(k-p+1)$-dimensional facet). For each $\phi \in \Phi$, let $F_{\mathcal{B}}(\phi)$ and $F_{\mathcal{B}'}(\phi)$ respectively denote the facets of minimal dimension which contain $\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B})$ and $\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}')$. Recall that for any basis $\mathcal{B}^*$ and any $\phi^* \in S_\phi$, every ℓ-dimensional facet of $\mathcal{C}(\phi^*, \mathcal{B}^*)$ is given by $cone(G(\phi^*)_{\bullet I})$ for some $I \subseteq \mathcal{B}^*$ with $|I| = \ell$. Hence, we have that there exist $J \subset \mathcal{B}$ and $J' \subset \mathcal{B}'$ such that $cone(G(\phi)_{\bullet J}) = F_{\mathcal{B}}(\phi)$ and $cone(G(\phi)_{\bullet J'}) = F_{\mathcal{B}'}(\phi)$ for all $\phi \in \Phi$. Thus, we also have that $k-p \leq |J| \leq k-p+1$, $k-p \leq |J'| \leq k-p+1$ and $dim(cone(G(\phi)_{\bullet J}) \cap cone(G(\phi)_{\bullet J'})) \geq k-p$. Furthermore, let $J^* = \begin{cases} J & \text{if } |J| \geq |J'| \\ J' & \text{otherwise} \end{cases}$ and recognize that we have either (i) $|J^*| = k - p$ and $span(G(\phi)_{\bullet J}) = span(G(\phi)_{\bullet J'})$ for all $\phi \in \Phi$, or (ii) $|J^*| = k - p + 1$ and $span(cone(G(\phi)_{\bullet J^*})) = span(cone(G(\phi)_{\bullet(J \cup J')}))$ for all $\phi \in \Phi$. More importantly, in either case we also have $aff(\mathcal{AS}(\phi)) \subseteq span(G(\phi)_{\bullet J^*})$ for all $\phi \in \Phi$. We now show that because $dim(\Phi) = p$, we actually have $aff(\mathcal{AS}(\phi)) \subseteq span(G(\phi)_{\bullet J^*})$ for all $\phi \in S_\phi$.

Recognize that because $aff(\mathcal{AS}(\phi)) \subseteq span(G(\phi)_{\bullet J^*})$ for all $\phi \in \Phi$, we have the following result.

$$\text{For every } \sigma \in \mathbb{R}^{k-p} \text{ and } \phi \in \Phi,\ \exists \chi^{\phi,\sigma} \in \mathbb{R}^{|J^*|} \text{ such that } q(\phi,\sigma) = G(\phi)_{\bullet J^*}\chi^{\phi,\sigma}. \tag{5.113}$$

Since $dim(\Phi) = p$, it is full dimensional in $\mathbb{R}^p$. Hence, there must exist $\epsilon > 0$ and $\phi' \in \Phi$ such that $B_\epsilon(\phi') \subseteq \Phi$. Thus, for every $\phi \in \mathbb{R}^p$ we have $\phi' + \epsilon\phi \in \Phi$. Thus, using (5.113) we obtain the following result.

$$q(\phi' + \epsilon\phi, \sigma) = G(\phi' + \epsilon\phi)_{\bullet J^*}\chi^{\phi' + \epsilon\phi,\sigma} \text{ for all } \phi \in \mathbb{R}^p \text{ and } \sigma \in \mathbb{R}^{k-p} \tag{5.114}$$

We now assume without loss of generality that $\phi' = \mathbf{0}$ and $\epsilon = 1$. If this were not the case, recognize that it could easily be achieved by constructing an equivalent mpLCP using a simple change of parameters in which we replace ϕ with $\frac{\tilde{\phi} - \phi'}{\epsilon}$. Under this assumption, (5.114) shows that for every $\phi \in \mathbb{R}^p$, each point in $\mathcal{AS}(\phi)$ can be represented as a linear combination of the columns of $G(\phi)_{\bullet J^*}$. Hence, $aff(\mathcal{AS}(\phi)) \subseteq span(G(\phi)_{\bullet J^*})$ for all $\phi \in \mathbb{R}^p \supset S_\phi$. □

Proposition 5.97. Let distinct complementary bases $\mathscr{B}$ and $\mathscr{B}'$ be given such that the invariancy regions $\mathcal{IR}_{\mathscr{B}}$ and $\mathcal{IR}_{\mathscr{B}'}$ overlap. Without loss of generality, there exists a partition of S_θ which does not include $\mathcal{IR}_{\mathscr{B}'}$.

Proof. The claim of the proposition is trivial when $\mathcal{IR}_{\mathscr{B}'} \subseteq \mathcal{IR}_{\mathscr{B}}$ and so we assume this is not the case. From the arguments used in the proof of Proposition 5.96, we can conclude that for every $\theta = (\phi, \sigma) \in \mathcal{IR}_{\mathscr{B}'}$, σ lies on a facet of $\mathcal{C}(\phi, \mathscr{B}')$ whose dimension is either $k - p$ or $k - p + 1$. Thus, since $M(\phi)$ is sufficient for each $\phi \in S_\phi$ and consequently $\mathcal{K}(M(\phi))$ is convex for each $\phi \in S_\phi$, for every $\theta = (\phi, \sigma) \in (\mathcal{IR}_{\mathscr{B}'} \setminus \mathcal{IR}_{\mathscr{B}})$ we must have that there exists a set $\mathcal{B}_\theta$ of complementary bases such that: (i) $\sigma \in \mathcal{C}(\phi, \mathscr{B}^*)$ for all $\mathscr{B}^* \in \mathcal{B}_\theta$, and (ii) $|\mathcal{B}_\theta \setminus \mathscr{B}'| \geq 1$. Furthermore, since there are a finite number of bases there must exist at least one basis $\mathscr{B}^\theta \in \mathcal{B}_\theta$ such that $dim(\mathcal{C}(\phi, \mathscr{B}^\theta) \cap \mathcal{AS}(\phi)) = k - p$ for all $\phi \in S_\phi$ such that $dim(\mathcal{C}(\phi, \mathscr{B}') \cap \mathcal{AS}(\phi)) = k - p$. Since $\mathcal{IR}_{\mathscr{B}'}$ is full dimensional, this shows that $\mathcal{IR}_{\mathscr{B}^\theta}$ is also full dimensional. Hence, we use the following strategy for including invariancy regions in our partition of S_θ: (i) Let $\mathscr{K} = \mathcal{IR}_{\mathscr{B}}$; (ii) Select a $\theta \in (\mathcal{IR}_{\mathscr{B}'} \setminus \mathscr{K})$; (iii) include $\mathcal{IR}_{\mathscr{B}^\theta}$

in the partition of S_θ and add $\mathcal{IR}_{\mathscr{B}^\theta}$ to $\mathscr{K}$; (iv) If $(\mathcal{IR}_{\mathscr{B}'} \setminus \mathscr{K}) \neq \emptyset$, go back to Step (ii). Thus, by following this strategy we ensure that, although $\mathcal{IR}_{\mathscr{B}'}$ is not included in the partition of S_θ, for every $\theta \in \mathcal{IR}_{\mathscr{B}'}$ there is a full dimensional invariancy region included in the partition of S_θ which contains θ. □

Recognize that result of Proposition 5.97 provides an answer to Question 5.93. Fortunately, the answer is that a true partition of S_θ exists whenever $M(\phi)$ is a sufficient matrix for all $\phi \in S_\phi$. Note that we have not studied whether or not the converse of this statement holds. The result we have obtained is certainly satisfactory, though, since we already assume in this work that $M(\phi)$ is sufficient for all $\phi \in S_\phi$.

We now move our focus to Question 5.94. It seems that the best strategy for ensuring that a true partition is discovered when attempting to partition S_θ is this: Each time a full dimensional invariancy region is being considered during phase 2, discard all other invariancy regions which overlap the one currently being considered. One very naive way in which this could be accomplished is given in the following steps.

1. Given a feasible complementary basis $\mathscr{B}$ for which $dim(\mathcal{IR}_{\mathscr{B}}) = k$, let $\mathscr{B} = \{\mathscr{B}\}$ and create a modified mpLCP is which S_θ is replaced by $\mathcal{IR}_{\mathscr{B}}$.

2. Perform phase 1 of our two phase mpLCP procedure. (Recall that $\mathscr{B}$ denotes the set of bases not considered in our procedure.)

3. If phase 1 returns an initial basis $\mathscr{B}'$ with $dim(\mathcal{IR}_{\mathscr{B}'}) = k$, then $dim(\mathcal{IR}_{\mathscr{B}})$ and $dim(\mathcal{IR}_{\mathscr{B}'})$ overlap. Add $\mathscr{B}'$ to $\mathscr{B}$ and repeat Step (2). Otherwise, if phase 1 does not return an initial basis, STOP.

This strategy then provides an answer to Question 5.94. We point out, though, that this may not be the most appropriate response to Question 5.94. It seems that a better response, although somewhat disappointing, should be that, even though one can ensure that a true partition can be obtained, it is generally impractical to do so. In the worst case the invariancy regions associated with every feasible complementary basis can overlap. To see this, recall Figure 5.1. In this figure, it could easily be the case that all parametric complementary cones, even those not depicted, could have a 1-dimensional intersection with $\mathcal{AS}(\phi)$ for all $\phi \in S_\phi$. Even though this result seems a bit disappointing, we do point out that obtaining a quasi-partition for mpLCP still provides a solution for every $\theta \in S_\theta$ and is therefore perfectly acceptable. Before ending this section, we do provide a few key results which establish situations in which we can be sure that overlapping regions either do not exist or will not be obtained by steps in our algorithms. We then provide one final result which offers a practical step that can be taken in order to eliminate from consideration some, though not all, bases whose invariancy regions overlap a region which is currently under consideration.

Proposition 5.98. For a complementary basis $\mathcal{B}$, if $\mathrm{Z}_{\mathcal{B}} = \emptyset$ then there does not exist another complementary basis $\mathcal{B}'$ such that $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ overlap.

Proof. Recognize that the set $\mathrm{Z}_{\mathcal{B}}$ can be interpreted as the set of $i \in \mathcal{B}$ such that for all $\phi \in S_\phi$, $G(\phi)_{\bullet i}$ is unnecessary in the representation of the points in $\mathcal{AS}(\phi)$ as linear combinations of the columns of $G(\phi)_{\bullet \mathcal{B}}$, i.e., $\mathcal{AS}(\phi) \subseteq span\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ for all $\phi \in S_\phi$. Thus, if $\mathrm{Z}_{\mathcal{B}} = \emptyset$ then $\mathcal{AS}(\phi)$ intersects the relative interior of $\mathcal{C}(\phi, \mathcal{B})$ for all but at most a finite number of $\phi \in S_\phi$. Hence, there cannot exist $\Phi \subseteq S_\phi$ such that $dim(\Phi) = p$ and $dim(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}')) = k - p$ for all $\phi \in \Phi$. Thus, the result of the proposition follows from Lemma 5.95. □

Proposition 5.99. Given a complementary basis $\mathcal{B}$ for which $dim(\mathcal{IR}_{\mathcal{B}}) = k$ and an $i \in \mathcal{B} \setminus \mathrm{Z}_{\mathcal{B}}$, neither a diagonal pivot involving i, as outlined in condition (1) of Proposition 5.72, nor an exchange pivot involving i, as outlined in condition (2) of Proposition 5.72, will result in a new basis $\mathcal{B}'$ for which $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ overlap.

Proof. Since $i \notin \mathrm{Z}_{\mathcal{B}}$ there are at most a finite number of $\phi \in S_\phi$ such that $G(\phi)_{\bullet i}$ is unnecessary in the representation of the points in $\mathcal{AS}(\phi)$ as linear combinations of the columns of $G(\phi)_{\bullet \mathcal{B}}$. This shows that there are at most a finite number of $\phi \in S_\phi$ such that $dim\left(\mathcal{AS}(\phi) \cap cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)\right) = k-p$, i.e., there is no $\Phi \subseteq S_\phi$ such that $dim(\Phi) = p$ and $dim\left(\mathcal{AS}(\phi) \cap cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)\right) = k - p$ for all $\phi \in \Phi$. Recall that if $\mathcal{B}'$ is obtained from $\mathcal{B}$ by a diagonal pivot involving i, as outlined in condition (1) of Proposition 5.72, or an exchange pivot involving i, as outlined in condition (2) of Proposition 5.72, then $(\mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}')) \subseteq cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ for all $\phi \in S_\phi$. Thus, there cannot exist a $\Phi \subseteq S_\phi$ such that $dim(\Phi) = p$ and $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}')\right) = k - p$ for all $\phi \in \Phi$. The result of the proposition then follows from Lemma 5.95. □

Proposition 5.100. Given a complementary basis $\mathcal{B}$ for which $dim(\mathcal{IR}_{\mathcal{B}}) = k$, if there exists an index $i \in \mathrm{Z}_{\mathcal{B}}$ such that $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero, then for the basis $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ we have that $\mathcal{IR}_{\mathcal{B}}$ and $\mathcal{IR}_{\mathcal{B}'}$ overlap.

Proof. Since $i \in \mathrm{Z}_{\mathcal{B}}$ we have $\mathcal{AS}(\phi) \subseteq span\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ for all $\phi \in S_{\phi}$. Furthermore, since $dim(\mathcal{IR}_{\mathcal{B}}) = k$ there must exist $\Phi \subset S_{\phi}$ such that $dim(\Phi) = p$ and $dim(\mathcal{AS}(\phi) \cap cone(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})})) = k - p$ for all $\phi \in \Phi$. Since $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero, $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is a basis and $\mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}') = cone\left(G(\phi)_{\bullet(\mathcal{B}\setminus\{i\})}\right)$ for all $\phi \in S_{\phi}$. Thus, $dim\left(\mathcal{AS}(\phi) \cap \mathcal{C}(\phi, \mathcal{B}) \cap \mathcal{C}(\phi, \mathcal{B}')\right) = k - p$ for all $\phi \in \Phi$. The result of the proposition then follows from Lemma 5.95. □

As a result of Proposition 5.100, the procedures we presented in Sections 5.4 and 5.5 can be modified so that whenever a basis $\mathcal{B}$ is discovered for which there exists an index $i \in \mathrm{Z}_{\mathcal{B}}$ such that $(T_{\mathcal{B}}(\phi))_{i,\bar{\imath}}$ is not identically zero, the basis $\mathcal{B}' = (\mathcal{B} \setminus \{i\}) \cup \{\bar{\imath}\}$ is added to $\mathscr{B}$ and therefore not considered for inclusion in the final partition of S_{θ}. The implementation we discuss in the next section incorporates this modification. We also point out that this modification is enough to ensure that the overlapping regions we discovered during our phase 1 consideration of Example 5.5 are not both considered. Observe from Table 5.4 in Section 5.A that $w_3 \in \mathrm{Z}^{ph1}_{\mathcal{B}^{*5.5}}$ and $(T^{ph1}_{\mathcal{B}^{*5.5}}(\phi))_{w_3,z_3}$ is not identically zero. Hence, we add $(\mathcal{B}^{*5.5} \setminus \{w_3\}) \cup \{z_3\} = \mathcal{B}^{5.5}_{iv}$ to $\mathscr{B}$ during the first iteration of the phase 1 algorithm and as a result, the invariancy region associated with this basis would not be considered in a later iteration of phase 1.

5.7 Experimental Results

We now present the results of a computational experiment we conducted in order to test the practical performance of the proposed algorithms. We also include a few brief notes on our implementation.

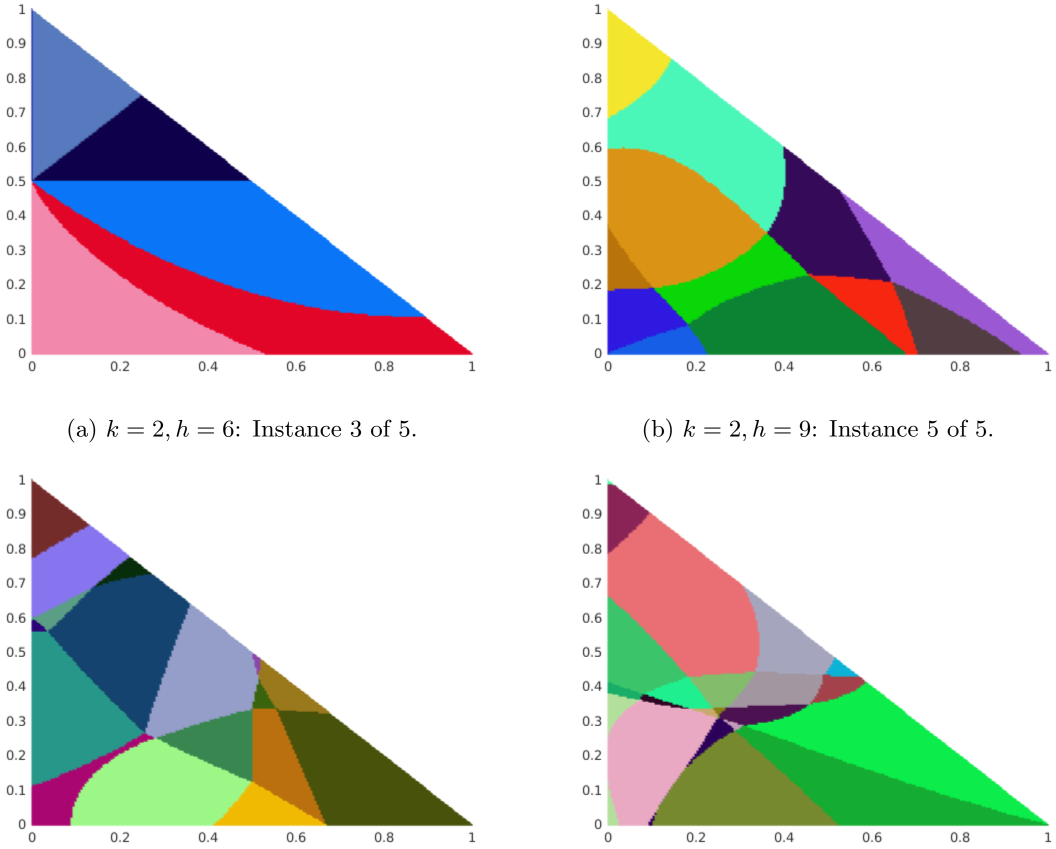
We implemented the proposed two-phase algorithm using MATLAB. All auxiliary NLPs were solved using the “fmincon” function within MATLAB. We note that since the version of mpLCP we consider in this work was previously unsolved, there is no other method with which we can compare. All tests were run using MATLAB R2016a [62] on a machine running Linux Mint 17 with two 2.4GHz processors, each with 4GB of RAM.

For our experiment we randomly generated 105 instances. We produced ten instances for each value of h in $\{4, \ldots, 12\}$; half with $k = 2$ and half with $k = 3$. We also produced an additional five instances for each value of h in $\{13, 14, 15\}$ with $k = 2$. Each instance was derived from a multiobjective program with $k+1$ convex quadratic objectives. These multiobjective programs were then scalarized using the weighted-sum method (see, for example, [27]) to obtain an mpQP in the form of (5.4), and then reformulated as an instance of mpLCP. We then solved each instance using our the implementation of the proposed method. A summary of the results is given in Table 5.3. As expected, the results display a positive correlation between instance size and numbers of iterations and regions as well as average CPU time spent in each iteration. Figure 5.6 depicts the partitions of S_θ computed during this experiment for four instances. Recall that for each pair of k and h values described in Table 5.3 we generated five instances of mpLCP. The label on each subfigure of Figure 5.6 indicates which of the five instances has its solution depicted in the figure.

Table 5.3: Experimental Results – Averages are taken over instances that were solved in under one hour.

k	h	Number Solved in < 1 hour	Average Time (s)	Average Num. Ph1 Iterations	Average Num. Ph2 Iterations	Average Num. Regions	Average Time Per Iteration (s)
2	4	5	7.1	1.2	3.0	1.8	1.6
	5	5	25.5	1.8	3.0	3.0	5.3
	6	5	50.7	2.0	6.6	4.8	5.8
	7	5	144.0	1.8	7.2	6.4	16.0
	8	5	66.5	2.0	4.2	4.2	10.7
	9	5	215.2	3.0	10.6	8.8	15.8
	10	5	132.4	2.0	6.8	5.8	15.0
	11	5	347.2	4.6	9.6	7.4	24.4
	12	4	460.8	2.2	12.2	10.0	31.7
	13	5	1,329.8	4.2	18.2	17.4	59.3
	14	4	816.5	3.0	13.7	12.2	48.0
	15	4	1,547.2	4.0	14.7	14.0	82.5
3	4	5	21.4	1.2	5.0	4.0	3.4
	5	5	47.7	3.6	5.0	4.4	5.5
	6	5	51.6	1.4	4.2	3.4	9.2
	7	5	212.0	1.6	8.6	8.0	20.7
	8	5	417.8	3.2	8.6	7.4	35.4
	9	5	1,406.8	2.8	15.8	12.0	75.6
	10	5	887.9	1.8	10.4	9.8	80.4
	11	4	2,124.4	2.0	14.2	11.7	130.7
	12	2	1,648.2	1.5	8.5	8.0	164.8

We now discuss a few details of our implementation. All parameters of the "fmincon" optimization function were left at their default values, except the constraint violation tolerance, which was set to 10^{-9}; the maximum number of iterations, which was set to $4,000$; and the maximum number of function evaluations, which was set to 8,000. Also, when solving NLPs as feasibility problems, we assumed that λ was sufficiently large when it reached a value of 10^{-4}. Additionally, in our implementation we explicitly compute the tableau associated with each discovered basis. Thus, the overall performance could likely be improved by instead using matrix factorization techniques. In the computational experiments conducted in [2], we discovered that the efficiency of phase 1 could be improved if we sorted k-dimensional boundaries of $(k+1)$-dimensional regions based the ρ



(a) $k = 2, h = 6$: Instance 3 of 5.

(b) $k = 2, h = 9$: Instance 5 of 5.

(c) $k = 2, h = 11$: Instance 4 of 5.

(d) $k = 2, h = 13$: Instance 1 of 5.

Figure 5.6: Partitions of S_θ.

component of the normalized normal vector of the boundaries (recall that in this work all boundaries are hyperplanes) prior to seeking adjacent regions across each boundary. We used a similar strategy here, but rather than sorting using the ρ component of the normal vector, we use the ρ component of the normalized gradient vector of each polynomial function defining a k-dimensional boundary of a given region, evaluated at the solution of NLP_s. By searching for adjacent regions across k-dimensional boundaries with the largest ρ component of this normalized gradient first, we increase the likelihood of discovering a new region that has a lower optimal value of NLP_s (minimal value of ρ) than the previous region. In this way we aim to speed up the process of discovering a region that intersects the hyperplane $\rho = 0$.

We also note that in our implementation, depending on the optimization technique chosen for use alongside MATLAB's "fmincon" function (interior point, trust region reflective, SQP, active set), the optimizer occasionally returns a local optimal solution which is not always feasible, even in situations in which a feasible solution exists. We found that this occurred more frequently for larger problems. To counter this issue, each time we use the “fmincon” function, we begin with SQP optimizer, but if the function fails or returns an infeasible solution, we resolve using the interior point optimizer. This is, of course, inefficient, and we will therefore seek a more robust optimizer to utilize in future implementations.

5.8 Conclusion

In this work we have introduced the first ever method for solving mpLCP (5.1) in which all elements of the matrix M and the vector q are permitted to be affine functions of the parameters, so long as $M(\theta)$ is a sufficient matrix for each permissible value of θ. Phase 1 answers the previously unanswered question of how one can determine an initial full dimensional invariancy region which can be used as a starting point in the process of partitioning the parameter space S_θ. The partition of S_θ is carried out in Phase 2. Experimental results are provided which give evidence of the utility of the proposed method.

In the future we will develop a more robust implementation of the procedures proposed in this work. We will also apply the presented methodology to several application problems which arise in areas of finance, engineering, health science, and more. As the methods we have presented can also be used to solve multiobjective (convex) quadratic optimization problems, we will also use the tools presented here as the foundation for a procedure for solving multiobjective mixed integer convex quadratic programs.

5.A Appendix A: Tableaux for Example 5.5

Table 5.4: $T^{ph1}_{\mathcal{B}^{*\,5.5}}(\phi,\rho)$

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	0	0	0	0	0	0	-1	-3	5	3
w_2	0	1	0	0	0	0	0	-2	-2	-2	$-\phi_1-2+3\rho$
w_3	0	0	1	0	0	1	2	$\phi_2-2\,\phi_1-4$	$2\,\phi_2-\phi_1-3$	$2-4\,\phi_2-3\,\phi_1$	0
w_4	0	0	0	1	0	3	2	$2\,\phi_2-\phi_1-3$	$\phi_1-\phi_2-4$	$3-4\,\phi_2-3\,\phi_1$	0
w_5	0	0	0	0	1	-5	2	$2-4\,\phi_2-3\,\phi_1$	$3-4\,\phi_2-3\,\phi_1$	ϕ_2-3	0

Table 5.5: $T^{ph1}_{\mathcal{B}_i^{5.5}}(\phi,\rho)$

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	$-\frac{1}{2}$	0	0	0	0	0	0	-2	6	$\frac{\phi_1}{2}-\frac{3\,\rho}{2}+4$
z_2	0	$\frac{\phi_2}{4}-\frac{\phi_1}{2}-1$	$\frac{1}{2}$	0	0	$\frac{1}{2}$	1	0	$\frac{\phi_1}{2}+\frac{\phi_2}{2}+\frac{1}{2}$	$3-\frac{5\,\phi_2}{2}-\frac{\phi_1}{2}$	$\frac{(2\,\phi_1-\phi_2+4)\left(\frac{\phi_1}{2}-\frac{3\,\rho}{2}+1\right)}{2}$
z_3	0	$-\frac{1}{2}$	0	0	0	0	0	1	1	1	$\frac{\phi_1}{2}-\frac{3\,\rho}{2}+1$
w_4	0	$\frac{\phi_1}{2}+\frac{\phi_2}{2}+\frac{1}{2}$	-1	1	0	2	0	0	$\phi_1-4\,\phi_2-2$	$-\phi_1-\phi_2$	$-\frac{(\phi_1+\phi_2+1)\,(\phi_1-3\,\rho+2)}{2}$
w_5	0	$3-\frac{5\,\phi_2}{2}-\frac{\phi_1}{2}$	-1	0	1	-6	0	0	$-\phi_1-\phi_2$	$4\,\phi_1+10\,\phi_2-11$	$\frac{(\phi_1+5\,\phi_2-6)\,(\phi_1-3\,\rho+2)}{2}$

Table 5.6: $T^{ph1}_{\mathcal{B}_{ii}^{5.5}}(\phi,\rho)$

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	$-\frac{3}{2}$	0	0	0	0	0	2	0	8	$\frac{3\,\phi_1}{2}-\frac{9\,\rho}{2}+6$
z_2	0	$\frac{\phi_1}{4}-\frac{\phi_2}{4}-1$	0	$\frac{1}{2}$	0	$\frac{3}{2}$	1	$\frac{3\,\phi_2}{2}-\phi_1+\frac{1}{2}$	0	$\frac{7}{2}-\frac{3\,\phi_2}{2}-2\,\phi_1$	$\frac{\left(\frac{\phi_1}{2}-\frac{3\,\rho}{2}+1\right)(\phi_2-\phi_1+4)}{2}$
w_3	0	$\frac{3\,\phi_2}{2}-\phi_1+\frac{1}{2}$	1	-1	0	-2	0	$\phi_1-4\,\phi_2-2$	0	$2\,\phi_1-3\,\phi_2-2$	$-\frac{(3\,\phi_2-2\,\phi_1+1)\,(\phi_1-3\,\rho+2)}{2}$
z_4	0	$-\frac{1}{2}$	0	0	0	0	0	1	1	1	$\frac{\phi_1}{2}-\frac{3\,\rho}{2}+1$
w_5	0	$\frac{7}{2}-\frac{3\,\phi_2}{2}-2\,\phi_1$	0	-1	1	-8	0	$2\,\phi_1-3\,\phi_2-2$	0	$7\,\phi_1+8\,\phi_2-13$	$\frac{(4\,\phi_1+3\,\phi_2-7)\,(\phi_1-3\,\rho+2)}{2}$

Table 5.7: $T^{ph1}_{\mathcal{B}^{5.5}_{iii}}(\phi, \rho)$

	w_1	w_2	w_3	w_4	w_5	z_1	z_2	z_3	z_4	z_5	
w_1	1	$\frac{5}{2}$	0	0	0	0	0	-6	-8	0	$\frac{15\,\rho}{2} - \frac{5\,\phi_1}{2} - 2$
z_2	0	$\frac{\phi_2}{4} - \frac{3}{4}$	0	0	$\frac{1}{2}$	$-\frac{5}{2}$	1	$\frac{5}{2} - \frac{5\,\phi_2}{2} - \frac{3\,\phi_1}{2}$	$3 - \frac{5\,\phi_2}{2} - \frac{3\,\phi_1}{2}$	0	$-\frac{(\phi_2-3)\left(\frac{\phi_1}{2}-\frac{3\,\rho}{2}+1\right)}{2}$
w_3	0	$\frac{5}{2} - \frac{5\,\phi_2}{2} - \frac{3\,\phi_1}{2}$	1	0	-1	6	0	$4\,\phi_1 + 10\,\phi_2 - 11$	$5\,\phi_1 + 11\,\phi_2 - 11$	0	$\frac{(3\,\phi_1+5\,\phi_2-5)\,(\phi_1-3\,\rho+2)}{2}$
w_4	0	$3 - \frac{5\,\phi_2}{2} - \frac{3\,\phi_1}{2}$	0	1	-1	8	0	$5\,\phi_1 + 11\,\phi_2 - 11$	$7\,\phi_1 + 8\,\phi_2 - 13$	0	$\frac{(3\,\phi_1+5\,\phi_2-6)\,(\phi_1-3\,\rho+2)}{2}$
z_5	0	$-\frac{1}{2}$	0	0	0	0	0	1	1	1	$\frac{\phi_1}{2} - \frac{3\,\rho}{2} + 1$

Table 5.8: $T^{ph1}_{\mathcal{B}^{5.5}_{iv}}(\phi, \rho)$ (Due to space limitations, we only display columns associated with nonbasic variables.)

	w_3	z_1	z_2	z_4	z_5	
w_1	$-\frac{1}{2\,\phi_1-\phi_2+4}$	$-\frac{1}{2\,\phi_1-\phi_2+4}$	$-\frac{2}{2\,\phi_1-\phi_2+4}$	$-\frac{5\,\phi_1-\phi_2+9}{2\,\phi_1-\phi_2+4}$	$\frac{13\,\phi_1-\phi_2+18}{2\,\phi_1-\phi_2+4}$	3
w_2	$-\frac{2}{2\,\phi_1-\phi_2+4}$	$-\frac{2}{2\,\phi_1-\phi_2+4}$	$-\frac{4}{2\,\phi_1-\phi_2+4}$	$-\frac{2\,\phi_1+2\,\phi_2+2}{2\,\phi_1-\phi_2+4}$	$\frac{2\,\phi_1+10\,\phi_2-12}{2\,\phi_1-\phi_2+4}$	$3\,\rho - \phi_1 - 2$
z_3	$-\frac{1}{2\,\phi_1-\phi_2+4}$	$-\frac{1}{2\,\phi_1-\phi_2+4}$	$-\frac{2}{2\,\phi_1-\phi_2+4}$	$\frac{\phi_1-2\,\phi_2+3}{2\,\phi_1-\phi_2+4}$	$\frac{3\,\phi_1+4\,\phi_2-2}{2\,\phi_1-\phi_2+4}$	0
w_4	$-\frac{\phi_1-2\,\phi_2+3}{2\,\phi_1-\phi_2+4}$	$\frac{5\,\phi_1-\phi_2+9}{2\,\phi_1-\phi_2+4}$	$\frac{2\,\phi_1+2\,\phi_2+2}{2\,\phi_1-\phi_2+4}$	$\frac{3\,\phi_1^2-7\,\phi_1\,\phi_2+2\,\phi_1+5\,\phi_2^2-12\,\phi_2-7}{2\,\phi_1-\phi_2+4}$	$-\frac{3\,\phi_1^2+7\,\phi_1\,\phi_2-\phi_1+4\,\phi_2^2+3\,\phi_2-6}{2\,\phi_1-\phi_2+4}$	0
w_5	$-\frac{3\,\phi_1+4\,\phi_2-2}{2\,\phi_1-\phi_2+4}$	$-\frac{13\,\phi_1-\phi_2+18}{2\,\phi_1-\phi_2+4}$	$-\frac{2\,\phi_1+10\,\phi_2-12}{2\,\phi_1-\phi_2+4}$	$-\frac{3\,\phi_1^2+7\,\phi_1\,\phi_2-\phi_1+4\,\phi_2^2+3\,\phi_2-6}{2\,\phi_1-\phi_2+4}$	$-\frac{-9\,\phi_1^2-26\,\phi_1\,\phi_2+18\,\phi_1-15\,\phi_2^2+9\,\phi_2+8}{2\,\phi_1-\phi_2+4}$	0

Table 5.9: $T^{ph1}_{\mathcal{B}_v^{5.5}}(\phi,\rho)$ (Due to space limitations, we only display columns associated with nonbasic variables.)

	w_2	w_3	w_4	z_1	z_5
w_1	$-\frac{\phi_1+6\,\phi_2+4}{2\,(4\,\phi_2-\phi_1+2)}$	$\frac{2}{4\,\phi_2-\phi_1+2}$	$-\frac{2}{4\,\phi_2-\phi_1+2}$	$-\frac{4}{4\,\phi_2-\phi_1+2}$	$\frac{2\,(\phi_1+\phi_2)}{4\,\phi_2-\phi_1+2}+6$
z_2	$\frac{\phi_2}{4}-\frac{\phi_1}{2}+\frac{\left(\frac{\phi_1}{2}+\frac{\phi_2}{2}+\frac{1}{2}\right)^2}{4\,\phi_2-\phi_1+2}-1$	$\frac{3\,\phi_2-2\,\phi_1+1}{2\,(4\,\phi_2-\phi_1+2)}$	$\frac{\phi_1+\phi_2+1}{8\,\phi_2-2\,\phi_1+4}$	$\frac{\phi_1+6\,\phi_2+4}{2\,(4\,\phi_2-\phi_1+2)}$	$-\frac{9\,\phi_1-13\,\phi_2+\phi_1\,\phi_2+21\,\phi_2^2-12}{2\,(4\,\phi_2-\phi_1+2)}$
z_3	$-\frac{3\,\phi_2-2\,\phi_1+1}{2\,(4\,\phi_2-\phi_1+2)}$	$-\frac{1}{4\,\phi_2-\phi_1+2}$	$\frac{1}{4\,\phi_2-\phi_1+2}$	$\frac{2}{4\,\phi_2-\phi_1+2}$	$1-\frac{\phi_1+\phi_2}{4\,\phi_2-\phi_1+2}$
z_4	$-\frac{\phi_1+\phi_2+1}{8\,\phi_2-2\,\phi_1+4}$	$\frac{1}{4\,\phi_2-\phi_1+2}$	$-\frac{1}{4\,\phi_2-\phi_1+2}$	$-\frac{2}{4\,\phi_2-\phi_1+2}$	$\frac{\phi_1+\phi_2}{4\,\phi_2-\phi_1+2}$
w_5	$-\frac{9\,\phi_1-13\,\phi_2+\phi_1\,\phi_2+21\,\phi_2^2-12}{2\,(4\,\phi_2-\phi_1+2)}$	$\frac{\phi_1+\phi_2}{4\,\phi_2-\phi_1+2}-1$	$-\frac{\phi_1+\phi_2}{4\,\phi_2-\phi_1+2}$	$-\frac{2\,(\phi_1+\phi_2)}{4\,\phi_2-\phi_1+2}-6$	$4\,\phi_1+10\,\phi_2+\frac{(\phi_1+\phi_2)^2}{4\,\phi_2-\phi_1+2}-11$

	RHS
w_1	$\frac{\phi_1}{2}-\frac{3\,\rho}{2}+\frac{(\phi_1+\phi_2+1)\,(\phi_1-3\,\rho+2)}{4\,\phi_2-\phi_1+2}+4$
z_2	$-\frac{(\phi_1-3\,\rho+2)\left(3\,\phi_1^2-7\,\phi_1\,\phi_2+2\,\phi_1+5\,\phi_2^2-12\,\phi_2-7\right)}{4\,(4\,\phi_2-\phi_1+2)}$
z_3	$\frac{(3\,\phi_2-2\,\phi_1+1)\,(\phi_1-3\,\rho+2)}{2\,(4\,\phi_2-\phi_1+2)}$
z_4	$\frac{(\phi_1+\phi_2+1)\,(\phi_1-3\,\rho+2)}{2\,(4\,\phi_2-\phi_1+2)}$
w_5	$\frac{(\phi_1-3\,\rho+2)\left(9\,\phi_1-13\,\phi_2+\phi_1\,\phi_2+21\,\phi_2^2-12\right)}{2\,(4\,\phi_2-\phi_1+2)}$

Table 5.10: $T^{ph1}_{\mathcal{B}^{5.5}_{vi}}(\phi, \rho)$ (Due to space limitations, we only display columns associated with nonbasic variables.)

	w_2	w_3	w_5	z_1	z_4
w_1	$\frac{2\,\phi_1+20\,\phi_2-25}{2\,(4\,\phi_1+10\,\phi_2-11)}$	$\frac{6}{4\,\phi_1+10\,\phi_2-11}$	$-\frac{6}{4\,\phi_1+10\,\phi_2-11}$	$\frac{36}{4\,\phi_1+10\,\phi_2-11}$	$-\frac{2\,(\phi_1+7\,\phi_2-11)}{4\,\phi_1+10\,\phi_2-11}$
z_2	$\frac{\phi_2}{4}-\frac{\phi_1}{2}-\frac{\left(\frac{\phi_1}{2}+\frac{5\,\phi_2}{2}-3\right)^2}{4\,\phi_1+10\,\phi_2-11}-1$	$\frac{3\,\phi_1+5\,\phi_2-5}{2\,(4\,\phi_1+10\,\phi_2-11)}$	$\frac{\phi_1+5\,\phi_2-6}{8\,\phi_1+20\,\phi_2-22}$	$-\frac{2\,\phi_1+20\,\phi_2-25}{2\,(4\,\phi_1+10\,\phi_2-11)}$	$\frac{3\,\phi_1^2+8\,\phi_1\,\phi_2-\phi_1+5\,\phi_2^2+5\,\phi_2-11}{2\,(4\,\phi_1+10\,\phi_2-11)}$
z_3	$-\frac{3\,\phi_1+5\,\phi_2-5}{2\,(4\,\phi_1+10\,\phi_2-11)}$	$\frac{1}{4\,\phi_1+10\,\phi_2-11}$	$-\frac{1}{4\,\phi_1+10\,\phi_2-11}$	$\frac{6}{4\,\phi_1+10\,\phi_2-11}$	$\frac{\phi_1+\phi_2}{4\,\phi_1+10\,\phi_2-11}+1$
w_4	$\frac{3\,\phi_1^2+8\,\phi_1\,\phi_2-\phi_1+5\,\phi_2^2+5\,\phi_2-11}{2\,(4\,\phi_1+10\,\phi_2-11)}$	$-\frac{\phi_1+\phi_2}{4\,\phi_1+10\,\phi_2-11}-1$	$\frac{\phi_1+\phi_2}{4\,\phi_1+10\,\phi_2-11}$	$\frac{2\,(\phi_1+7\,\phi_2-11)}{4\,\phi_1+10\,\phi_2-11}$	$\phi_1-4\,\phi_2-\frac{(\phi_1+\phi_2)^2}{4\,\phi_1+10\,\phi_2-11}-2$
z_5	$-\frac{\phi_1+5\,\phi_2-6}{8\,\phi_1+20\,\phi_2-22}$	$-\frac{1}{4\,\phi_1+10\,\phi_2-11}$	$\frac{1}{4\,\phi_1+10\,\phi_2-11}$	$-\frac{6}{4\,\phi_1+10\,\phi_2-11}$	$-\frac{\phi_1+\phi_2}{4\,\phi_1+10\,\phi_2-11}$

	RHS
w_1	$\frac{\phi_1}{2}-\frac{3\,\rho}{2}-\frac{3\,(\phi_1+5\,\phi_2-6)\,(\phi_1-3\,\rho+2)}{4\,\phi_1+10\,\phi_2-11}+4$
z_2	$-\frac{(\phi_1-3\,\rho+2)\left(-9\,\phi_1^2-26\,\phi_1\,\phi_2+18\,\phi_1-15\,\phi_2^2+9\,\phi_2+8\right)}{4\,(4\,\phi_1+10\,\phi_2-11)}$
z_3	$\frac{(3\,\phi_1+5\,\phi_2-5)\,(\phi_1-3\,\rho+2)}{2\,(4\,\phi_1+10\,\phi_2-11)}$
w_4	$\frac{(\phi_1+\phi_2)\,(\phi_1+5\,\phi_2-6)\,(\phi_1-3\,\rho+2)}{2\,(4\,\phi_1+10\,\phi_2-11)}-\frac{(\phi_1+\phi_2+1)\,(\phi_1-3\,\rho+2)}{2}$
z_5	$\frac{(\phi_1+5\,\phi_2-6)\,(\phi_1-3\,\rho+2)}{2\,(4\,\phi_1+10\,\phi_2-11)}$

Table 5.11: $T^{ph1}_{\mathcal{B}_0^{5.5}}(\phi,\rho)$. Note that this tableau also serves as the first tableau for phase 2 of Example 5.5, by substituting $\rho = 0$. (Due to space limitations, we only display columns associated with nonbasic variables.)

	w_2	w_3	w_4
w_1	$-\frac{3\phi_1^2+18\phi_1\phi_2-37\phi_1-75\phi_2^2+64\phi_2+28}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$	$-\frac{2(\phi_1-20\phi_2+5)}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$-\frac{2\phi_1+14\phi_2-22}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$
z_2	$\frac{9\phi_1^3-9\phi_1^2\phi_2-33\phi_1^2-87\phi_1\phi_2^2+21\phi_1\phi_2+22\phi_1-59\phi_2^3+13\phi_2^2+50\phi_2+5}{4\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$	$\frac{-6\phi_1^2-\phi_1\phi_2+11\phi_1+15\phi_2^2-16\phi_2+1}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$	$\frac{3\phi_1^2+8\phi_1\phi_2-\phi_1+5\phi_2^2+5\phi_2-11}{-6\phi_1^2+16\phi_1\phi_2+38\phi_1+82\phi_2^2-48\phi_2-44}$
z_3	$-\frac{-6\phi_1^2-\phi_1\phi_2+11\phi_1+15\phi_2^2-16\phi_2+1}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$	$-\frac{7\phi_1+8\phi_2-13}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$\frac{5\phi_1+11\phi_2-11}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$
z_4	$-\frac{3\phi_1^2+8\phi_1\phi_2-\phi_1+5\phi_2^2+5\phi_2-11}{-6\phi_1^2+16\phi_1\phi_2+38\phi_1+82\phi_2^2-48\phi_2-44}$	$\frac{5\phi_1+11\phi_2-11}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$-\frac{4\phi_1+10\phi_2-11}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$
z_5	$-\frac{9\phi_1-13\phi_2+\phi_1\phi_2+21\phi_2^2-12}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$	$-\frac{3\phi_2-2\phi_1+2}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$-\frac{\phi_1+\phi_2}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$

	w_5	z_1
w_1	$-\frac{2(13\phi_2-2\phi_1+6)}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$\frac{4(32\phi_2-7\phi_1+29)}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$
z_2	$\frac{9\phi_1-13\phi_2+\phi_1\phi_2+21\phi_2^2-12}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$	$\frac{3\phi_1^2+18\phi_1\phi_2-37\phi_1-75\phi_2^2+64\phi_2+28}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$
z_3	$-\frac{3\phi_2-2\phi_1+2}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$-\frac{2(\phi_1-20\phi_2+5)}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$
z_4	$-\frac{\phi_1+\phi_2}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$-\frac{2\phi_1+14\phi_2-22}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$
z_5	$\frac{4\phi_2-\phi_1+2}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$	$-\frac{2(13\phi_2-2\phi_1+6)}{-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22}$

	RHS
w_1	$-\frac{16\phi_2-68\phi_1+84\rho-148\phi_1\phi_2-111\phi_1\rho+192\phi_2\rho+75\phi_1\phi_2^2-18\phi_1^2\phi_2+9\phi_1^2\rho-225\phi_2^2\rho+49\phi_1^2-3\phi_1^3-96\phi_2^2+54\phi_1\phi_2\rho+76}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$
z_2	$-\frac{(\phi_1-3\rho+2)\left(9\phi_1^3-9\phi_1^2\phi_2-33\phi_1^2-87\phi_1\phi_2^2+21\phi_1\phi_2+22\phi_1-59\phi_2^3+13\phi_2^2+50\phi_2+5\right)}{4\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$
z_3	$\frac{(\phi_1-3\rho+2)\left(-6\phi_1^2-\phi_1\phi_2+11\phi_1+15\phi_2^2-16\phi_2+1\right)}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$
z_4	$\frac{(\phi_1-3\rho+2)\left(3\phi_1^2+8\phi_1\phi_2-\phi_1+5\phi_2^2+5\phi_2-11\right)}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$
z_5	$\frac{(\phi_1-3\rho+2)\left(9\phi_1-13\phi_2+\phi_1\phi_2+21\phi_2^2-12\right)}{2\left(-3\phi_1^2+8\phi_1\phi_2+19\phi_1+41\phi_2^2-24\phi_2-22\right)}$

Table 5.12: $T_{\mathscr{B}_1^{5.5}}(\phi)$ (Due to space limitations, we only display columns associated with nonbasic variables.)

	w_1	w_2	w_3	w_4	w_5
z_1	$\frac{-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22}{4\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{3\,\phi_1^2+18\,\phi_1\,\phi_2-37\,\phi_1-75\,\phi_2^2+64\,\phi_2+28}{8\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{\phi_1-20\,\phi_2+5}{2\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{\phi_1+7\,\phi_2-11}{64\,\phi_2-14\,\phi_1+58}$	$-\frac{13\,\phi_2-2\,\phi_1+6}{2\,(32\,\phi_2-7\,\phi_1+29)}$
z_2	$-\frac{3\,\phi_1^2+18\,\phi_1\,\phi_2-37\,\phi_1-75\,\phi_2^2+64\,\phi_2+28}{8\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{-81\,\phi_1^2+288\,\phi_1\,\phi_2+69\,\phi_1+47\,\phi_2^2+388\,\phi_2+62}{16\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{29\,\phi_1-60\,\phi_2+9}{4\,(32\,\phi_2-7\,\phi_1+29)}$	$\frac{13\,\phi_1-5\,\phi_2+43}{4\,(32\,\phi_2-7\,\phi_1+29)}$	$\frac{2\,\phi_1+9\,\phi_2+24}{4\,(32\,\phi_2-7\,\phi_1+29)}$
z_3	$\frac{\phi_1-20\,\phi_2+5}{2\,(32\,\phi_2-7\,\phi_1+29)}$	$\frac{29\,\phi_1-60\,\phi_2+9}{4\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{16}{32\,\phi_2-7\,\phi_1+29}$	$\frac{12}{32\,\phi_2-7\,\phi_1+29}$	$\frac{4}{32\,\phi_2-7\,\phi_1+29}$
z_4	$\frac{\phi_1+7\,\phi_2-11}{64\,\phi_2-14\,\phi_1+58}$	$-\frac{13\,\phi_1-5\,\phi_2+43}{4\,(32\,\phi_2-7\,\phi_1+29)}$	$\frac{12}{32\,\phi_2-7\,\phi_1+29}$	$-\frac{9}{32\,\phi_2-7\,\phi_1+29}$	$-\frac{3}{32\,\phi_2-7\,\phi_1+29}$
z_5	$\frac{13\,\phi_2-2\,\phi_1+6}{2\,(32\,\phi_2-7\,\phi_1+29)}$	$-\frac{2\,\phi_1+9\,\phi_2+24}{4\,(32\,\phi_2-7\,\phi_1+29)}$	$\frac{4}{32\,\phi_2-7\,\phi_1+29}$	$-\frac{3}{32\,\phi_2-7\,\phi_1+29}$	$-\frac{1}{32\,\phi_2-7\,\phi_1+29}$

	RHS
z_1	$\frac{3\,\phi_1^3+18\,\phi_1^2\,\phi_2-49\,\phi_1^2-75\,\phi_1\,\phi_2^2+148\,\phi_1\,\phi_2+68\,\phi_1+96\,\phi_2^2-16\,\phi_2-76}{8\,(32\,\phi_2-7\,\phi_1+29)}$
z_2	$\frac{-81\,\phi_1^3+288\,\phi_1^2\,\phi_2-111\,\phi_1^2+47\,\phi_1\,\phi_2^2+856\,\phi_1\,\phi_2+422\,\phi_1+544\,\phi_2^2+392\,\phi_2-44}{16\,(32\,\phi_2-7\,\phi_1+29)}$
z_3	$-\frac{61\,\phi_1-60\,\phi_1\,\phi_2+29\,\phi_1^2-12}{4\,(32\,\phi_2-7\,\phi_1+29)}$
z_4	$\frac{75\,\phi_1+32\,\phi_2-5\,\phi_1\,\phi_2+13\,\phi_1^2+20}{4\,(32\,\phi_2-7\,\phi_1+29)}$
z_5	$\frac{16\,\phi_1+96\,\phi_2+9\,\phi_1\,\phi_2+2\,\phi_1^2+84}{4\,(32\,\phi_2-7\,\phi_1+29)}$

Table 5.13: $T_{\mathscr{B}_2^{5.5}}(\phi)$ (Due to space limitations, we only display columns associated with nonbasic variables.)

	w_2	w_4	w_5	z_1	z_3
w_1	$\frac{11\,\phi_1-17}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$\frac{8}{7\,\phi_1+8\,\phi_2-13}$	$-\frac{8}{7\,\phi_1+8\,\phi_2-13}$	$\frac{64}{7\,\phi_1+8\,\phi_2-13}$	$-\frac{2\,(\phi_1-20\,\phi_2+5)}{7\,\phi_1+8\,\phi_2-13}$
z_2	$\frac{-9\,\phi_1^2-23\,\phi_1\,\phi_2+15\,\phi_1-17\,\phi_2^2+23\,\phi_2+3}{4\,(7\,\phi_1+8\,\phi_2-13)}$	$\frac{3\,\phi_1+5\,\phi_2-6}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$\frac{4\,\phi_1+3\,\phi_2-7}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$-\frac{11\,\phi_1-17}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$\frac{-6\,\phi_1^2-\phi_1\,\phi_2+11\,\phi_1+15\,\phi_2^2-16\,\phi_2+1}{2\,(7\,\phi_1+8\,\phi_2-13)}$
w_3	$\frac{-6\,\phi_1^2-\phi_1\,\phi_2+11\,\phi_1+15\,\phi_2^2-16\,\phi_2+1}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$-\frac{5\,\phi_1+11\,\phi_2-11}{7\,\phi_1+8\,\phi_2-13}$	$\frac{3\,\phi_2-2\,\phi_1+2}{7\,\phi_1+8\,\phi_2-13}$	$\frac{2\,(\phi_1-20\,\phi_2+5)}{7\,\phi_1+8\,\phi_2-13}$	$-\frac{-3\,\phi_1^2+8\,\phi_1\,\phi_2+19\,\phi_1+41\,\phi_2^2-24\,\phi_2-22}{7\,\phi_1+8\,\phi_2-13}$
z_4	$-\frac{3\,\phi_1+5\,\phi_2-6}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$\frac{1}{7\,\phi_1+8\,\phi_2-13}$	$-\frac{1}{7\,\phi_1+8\,\phi_2-13}$	$\frac{8}{7\,\phi_1+8\,\phi_2-13}$	$\frac{5\,\phi_1+11\,\phi_2-11}{7\,\phi_1+8\,\phi_2-13}$
z_5	$-\frac{4\,\phi_1+3\,\phi_2-7}{2\,(7\,\phi_1+8\,\phi_2-13)}$	$-\frac{1}{7\,\phi_1+8\,\phi_2-13}$	$\frac{1}{7\,\phi_1+8\,\phi_2-13}$	$-\frac{8}{7\,\phi_1+8\,\phi_2-13}$	$-\frac{3\,\phi_2-2\,\phi_1+2}{7\,\phi_1+8\,\phi_2-13}$

	RHS
w_1	$\frac{-11\,\phi_1^2+37\,\phi_1+48\,\phi_2-44}{2\,(7\,\phi_1+8\,\phi_2-13)}$
z_2	$-\frac{(\phi_1+2)\left(-9\,\phi_1^2-23\,\phi_1\,\phi_2+15\,\phi_1-17\,\phi_2^2+23\,\phi_2+3\right)}{4\,(7\,\phi_1+8\,\phi_2-13)}$
w_3	$-\frac{(\phi_1+2)\left(-6\,\phi_1^2-\phi_1\,\phi_2+11\,\phi_1+15\,\phi_2^2-16\,\phi_2+1\right)}{2\,(7\,\phi_1+8\,\phi_2-13)}$
z_4	$\frac{(\phi_1+2)\,(3\,\phi_1+5\,\phi_2-6)}{2\,(7\,\phi_1+8\,\phi_2-13)}$
z_5	$\frac{(\phi_1+2)\,(4\,\phi_1+3\,\phi_2-7)}{2\,(7\,\phi_1+8\,\phi_2-13)}$

5.B Appendix B: Tableaux for Example 5.6

Table 5.14: $T_{\mathcal{B}_0^{5.6}}(\phi, \sigma)$

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	
w_1	1	0	0	0	0	0	2	1	$-\sigma - 1$
w_2	0	1	0	0	0	0	5	$-\phi - 7$	$\phi - \sigma - 1$
w_3	0	0	1	0	-1	-3	0	0	$-18\,\sigma - 34$
w_4	0	0	0	1	-1	$\phi + 5$	0	0	$-9\,\sigma - 17$

Table 5.15: $T_{\mathcal{B}_1^{5.6}}(\phi, \sigma)$

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	
w_1	1	$\frac{1}{\phi+7}$	0	0	0	0	$\frac{5}{\phi+7} + 2$	0	$-\frac{8\,\sigma+\phi\,\sigma+8}{\phi+7}$
z_2	0	0	0	$\frac{1}{\phi+5}$	$-\frac{1}{\phi+5}$	1	0	0	$-\frac{9\,\sigma+17}{\phi+5}$
w_3	0	0	1	$\frac{3}{\phi+5}$	$-\frac{3}{\phi+5} - 1$	0	0	0	$-\frac{(2\,\phi+13)\,(9\,\sigma+17)}{\phi+5}$
z_4	0	$-\frac{1}{\phi+7}$	0	0	0	0	$-\frac{5}{\phi+7}$	1	$\frac{\sigma-\phi+1}{\phi+7}$

Table 5.16: $T_{\mathcal{B}_2^{5.6}}(\phi, \sigma)$

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	
w_1	1	$-\frac{2}{5}$	0	0	0	0	0	$\frac{2\,\phi}{5} + \frac{19}{5}$	$-\frac{2\,\phi}{5} - \frac{3\,\sigma}{5} - \frac{3}{5}$
z_2	0	0	$-\frac{1}{3}$	0	$\frac{1}{3}$	1	0	0	$6\,\sigma + \frac{34}{3}$
z_3	0	$\frac{1}{5}$	0	0	0	0	1	$-\frac{\phi}{5} - \frac{7}{5}$	$\frac{\phi}{5} - \frac{\sigma}{5} - \frac{1}{5}$
w_4	0	0	$\frac{\phi}{3} + \frac{5}{3}$	1	$-\frac{\phi}{3} - \frac{8}{3}$	0	0	0	$-\frac{(2\,\phi+13)\,(9\,\sigma+17)}{3}$

Table 5.17: $T_{\mathcal{B}_3^{5.6}}(\phi, \sigma)$

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	
z_1	0	0	$-\frac{\phi+5}{\phi+8}$	$-\frac{3}{\phi+8}$	1	0	0	0	$\frac{(2\,\phi+13)\,(9\,\sigma+17)}{\phi+8}$
z_2	0	0	$-\frac{1}{\phi+8}$	$\frac{1}{\phi+8}$	0	1	0	0	$\frac{9\,\sigma+17}{\phi+8}$
z_3	$\frac{\phi+7}{2\,\phi+19}$	$\frac{1}{2\,\phi+19}$	0	0	0	0	1	0	$-\frac{8\,\sigma+\phi\,\sigma+8}{2\,\phi+19}$
z_4	$\frac{5}{2\,\phi+19}$	$-\frac{2}{2\,\phi+19}$	0	0	0	0	0	1	$-\frac{2\,\phi+3\,\sigma+3}{2\,\phi+19}$

Table 5.18: $T_{\mathcal{B}_4^{5.6}}(\phi, \sigma)$

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	
z_1	0	0	0	-1	1	$-\phi - 5$	0	0	$9\,\sigma + 17$
w_2	$\phi + 7$	1	0	0	0	0	$2\,\phi + 19$	0	$-8\,\sigma - \phi\,\sigma - 8$
w_3	0	0	1	-1	0	$-\phi - 8$	0	0	$-9\,\sigma - 17$
z_4	1	0	0	0	0	0	2	1	$-\sigma - 1$

Table 5.19: $T_{\mathcal{B}_5^{5.6}}(\phi, \sigma)$

	w_1	w_2	w_3	w_4	z_1	z_2	z_3	z_4	
z_1	0	0	-1	0	1	3	0	0	$18\,\sigma + 34$
w_2	$-\frac{5}{2}$	1	0	0	0	0	0	$-\phi - \frac{19}{2}$	$\phi + \frac{3\,\sigma}{2} + \frac{3}{2}$
z_3	$\frac{1}{2}$	0	0	0	0	0	1	$\frac{1}{2}$	$-\frac{\sigma}{2} - \frac{1}{2}$
w_4	0	0	-1	1	0	$\phi + 8$	0	0	$9\,\sigma + 17$

Chapter 6

Conclusion and Future Research

In this final chapter of the dissertation we first provide a summary of the contributions to the field of Operations Research presented in this work, and then discuss directions of future research.

6.1 Summary of Contributions

In this work we have presented significant contributions in a variety of areas of mathematical programming by introducing new theory and methodology which can be used to solve specific classes of multiobjective and parametric programs. We have also provided empirical evidence of the practical relevance of these contributions by implementing each of the proposed techniques and conducting computational experiments comparing the performance of these techniques with that of current state-of-the-art procedures.

6.1.1 Primary Theoretical Contributions

In Chapter 4 we uncover various aspects of the structure of mpLCP with parameters present in the q vector which were previously unknown, particularly for degenerate problems. Chapter 5 contains perhaps our most important theoretical contribution. Here we conduct the first ever study of mpLCP with parameters in general positions within the M matrix. We reveal that the parameter space S_θ can be partitioned into a set of regions such that within each region the representation of the decision variables as functions of the parameters is invariant. Moreover, we discover that each of these regions can be represented as a semi-algebraic set, i.e., a set defined by polynomial inequalities. As a result, we are first to study this problem in the context of algebraic geometry. We unveil many other fascinating aspects of the problem's structure and introduce breakthrough theory which allows us to present the first procedure for solving this problem.

6.1.2 Primary Methodological/Computational Contributions

In Chapter 2 we introduce a new data structure, in the form of a modified binary tree, that is able to efficiently store sets of nondominated solutions of BOMILPs. We conduct three computational experiments aimed at testing the practical value of the proposed data structure. The results show that this structure provides a more efficient method for storing solutions to BOMILP than other prevalent techniques and that the new data structure is quite useful when when paired with branch-and-bound methods for solving BOMILPs.

In Chapter 3 we provide the first ever study of presolve techniques for BOMILP. We present justification for the extension of several presolve processes commonly used alongside single objective BB procedures to the biobjective setting. We also give the first comprehensive study of a branch-and-bound procedure designed for solving BOMILPs with general integers. We provide a new interpretation of bound sets, new techniques for checking previously proposed fathoming rules, and introduce new fathoming rules. We are also first to discuss the extension of procedures for presolve, preprocessing, branching, and the measurement of a duality gap to the context of general BOMILP. The computational experiments we performed show that the BB scheme we present is extremely useful and outperforms current state-of-the-art techniques for solving BOMILP for all problems previously considered in the literature. We therefore also present a new set of challenging BOMILP instances adapted from practical instances of single objective mixed-integer linear programs available in the MIPlib library. Computational tests performed on these instances indicate that the new BB procedure still performs comparably with the current state-of-the-art even for significantly challenging problems.

Chapters 4 and 5 contain studies of mpLCP in which we develop new techniques for partitioning the parameter space $S_\theta \subset \mathbb{R}^k$ into a set of so-called invariancy regions such that, within each region the representation of the decision variables as functions of the parameters is invariant at the mpLCP solution. For the study contained in Chapter 4 we allow parameters only within the q vector. Hence, the problem considered in this chapter is a reduced version of the problem considered in Chapter 5, where we permit parameters within both the q vector and the M matrix. The methods presented in each of these chapters are two phase procedures. In Phase 1 of each procedure an initial region of full dimension with which one can begin a procedure for partitioning S_θ is obtained. In phase 2, given any invariancy region which is k or $(k-1)$-dimensional, all $(k-1)$-dimensional boundaries of the invariancy region are determined, and all other invariancy regions whose dimesions are at least $k-1$ and which are adjacent to the original invariancy region across each of its $(k-1)$-dimensional boundaries are computed. Hence, at termination of phase 2 the set of discovered k-dimensional invariancy regions forms a partition of the feasible subset of S_θ. The worst-case complexity of the method presented in Chapter 4 is one order of magnitude lower than that of the current state-of-the-art procedure for degenerate problems and matches that of the current state-of-the-art procedure for nondegenerate problems. Additionally, experimental results were conducted in which we compared the performance of the two phase method proposed in Chapter 4 with the current state-of-the-art procedure. The results showed that our method performs significantly better, often achieving running times two orders of magnitude faster than the current state-of-the-art procedure. The problem considered in Chapter 5 was previously unsolved. Nevertheless, we implemented the proposed two phase procedure and provide the results of of its performance on a set of reasonably sized, randomly generated instances.

6.2 Future Research

The work we have done in this dissertation leads to several new and exciting areas in which we may conduct research in the future. Some of these areas can be studied immediately, but some will require the construction of additional building blocks before they can be studied extensively. For this reason, we separate our discussion of future research into short term and long term goals.

6.2.1 Short Term Goals for Future Research

The studies we have conducted on BOMILP have revealed that, although there is much potential for the development of highly efficient and effective branch-and-bound procedures for BOMILP, these procedures are extremely complex and quite difficult to implement efficiently using current commercial optimization packages. It seems that in order to develop a truly efficient implementation of a BB procedure for BOMILP one must either opt to build his own solver without the use of commercial packages, or wait until tools for solving BOMILP to be incorporated within commercial software. Unfortunately, neither of these options is very promising. We have seen, however, that there is some benefit to incorporating various aspects of objective space search procedures, such as the triangle splitting method, alongside branch-and-bound methods. For this reason, in the near future we will study objective space search procedures in more detail, and develop a hybrid solution procedure for solving BOMILP which combines aspects of both branch-and-bound procedures and objective space search methods.

The LCP is a highly applicable problem, especially since it encompasses linear programs and convex quadratic programs. By extension, mpLCP then encompasses multiparametric linear programs and convex quadratic programs. As our study of mpLCP was the first to consider the case in which parameters are permitted in general locations within the M matrix, the solution procedure we presented can be applied to a variety of applications from disciplines such as finance, business and the various areas of engineering, which can be modelled as either multiobjective programs or problems of optimization under uncertainty and have yet been unstudied. Hence, in the near future we will collaborate with other researchers from each of these disciplines in order to discover interesting applications to which we can apply the new methodology.

6.2.2 Long Term Goals for Future Research

In the studies we have conducted for BOMILP, we have found the computation of nondominated subsets of solutions to be sufficiently challenging even in the presence of only two objectives. Nevertheless, there is significant need to extend the ideas we have developed to the case of three or more objectives. We will study ways in which this can be done, and use our findings to develop methods for solving multiobjective mixed-integer linear programs with more than two objectives.

The methods we have developed for solving mpLCP can be used to solve multiobjective optimization problems with any number of convex quadratic and linear objectives and linear constraints. As a result, the presented solution procedure serves as a sufficient extension of the parametric simplex method to the context of multiobjective convex quadratic programs. Thus, in future work we will use the mpLCP method we have presented as a foundational tool with which we will develop a method for solving biobjective, and eventually multiobjective, mixed-integer quadratic programs.

Bibliography

[1] N. Adelgren and A. Gupte. A branch-and-bound for multiobjective MIPs and its implementation for biobjective problems. *Working Paper*, 2016. URL `http://agupte.people.clemson.edu/MultiobjBB.pdf`.

[2] N. Adelgren and M. M. Wiecek. A two-phase algorithm for the multiparametric linear complementarity problem. *European Journal of Operational Research*, 254(3):715–738, 2016.

[3] N. Adelgren, P. Belotti, and A. Gupte. Efficient storage of Pareto points in biobjective mixed integer programming. *Under Review*, 2014. URL `http://arxiv.org/abs/1411.6538`.

[4] S. R. Arora, R. Shanker, and N. Malhotra. Enumeration technique for solving multi-objective quadratic set-covering problem using goal programming. *Opsearch*, 48(1):20–29, 2011.

[5] M. Baotić. An efficient algorithm for multi-parametric quadratic programming. *Report AUT02-05, Institut für Automatik, ETH Zürich*, 2002.

[6] S. Barnett. A simple class of parametric linear programming problems. *Operations Research*, 16(6):1160–1165, 1968.

[7] A. Barvinok. *A Course in Convexity*, volume 54. American Mathematical Society, 2002.

[8] M. S. Bazaraa, H. D. Sherali, and C. Shetty. *Nonlinear Programming: Theory and Algorithms.* John Wiley & Sons, 2006.

[9] A. Beato-Moreno, P. Luque-Calvo, R. Osuna-Gómez, and A. Rufián-Lizana. Finding efficient points in multiobjective quadratic programming with strictly convex objective functions. In *Advances in multiple objective and goal programming (Torremolinos, 1996)*, volume 455 of *Lecture Notes in Econom. and Math. Systems*, pages 365–373. Springer, Berlin, 1997.

[10] P. Belotti, B. Soylu, and M. M. Wiecek. A branch-and-bound algorithm for biobjective mixed-integer programs. *Technical Report, Clemson University*, 2013. URL `http://www.clemson.edu/ces/math/technical_reports/belotti.bb-bicriteria.pdf`.

[11] P. Belotti, B. Soylu, and M. M. Wiecek. Fathoming rules for biobjective mixed-integer linear programs. *Technical Report, Clemson University*, November 2015. URL `http://www.clemson.edu/ces/departments/math/documents/technical-reports/TR2015_11_pb.bs.mw.pdf`.

[12] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit solution of model predictive control via multiparametric quadratic programming. In *Proceedings of the 2000 American Control Conference*, volume 2, pages 872–876. IEEE, 2000.

[13] M. Bôcher and E. P. R. Duval. *Introduction to Higher Algebra.* Macmillan, 1907.

[14] J. Bochnak, M. Coste, and M.-F. Roy. *Real Algebraic Geometry*, volume 36. Springer Science & Business Media, 2013.

[15] N. Boland, H. Charkhgard, and M. Savelsbergh. The triangle splitting method for biobjective mixed integer programming. In J. Lee and J. Vygen, editors, *Integer Programming and Combinatorial Optimization (IPCO)*, volume 8494 of *Lecture Notes in Computer Science*, pages 162–173. Springer, 2014.

[16] N. Boland, H. Charkhgard, and M. Savelsbergh. A criterion space search algorithm for biobjective mixed integer programming: The triangle splitting method. *INFORMS Journal on Computing*, 27(4):597–618, 2015.

[17] B. Chakraborty, S. Nanda, and M. Biswal. On the solution of parametric linear complementarity problems. *International Journal of Pure and Applied Mathematics*, 17:9–18, 2004.

[18] S. Columbano, K. Fukuda, and C. N. Jones. An output-sensitive algorithm for multiparametric LCPs with sufficient matrices. In *Polyhedral Computation*, volume 48, pages 73–102. American Mathematical Society, 2009.

[19] R. Cottle, J.-S. Pang, and V. Venkateswaran. Sufficient matrices and the linear complementarity problem. *Linear Algebra and Its Applications*, 114:231–249, 1989.

[20] R. W. Cottle. Monotone solutions of the parametric linear complementarity problem. *Mathematical Programming*, 3(1):210–224, 1972.

[21] R. W. Cottle. A field guide to the matrix classes found in the literature of the linear complementarity problem. *Journal of Global Optimization*, 46(4):571–580, 2010.

[22] R. W. Cottle, J.-S. Pang, and R. E. Stone. *The Linear Complementarity Problem.* SIAM, 2009.

[23] M. Courtillot. On varying all the parameters in a linear-programming problem and sequential solution of a linear-programming problem. *Operations Research*, 10(4):471–475, 1962.

[24] R. Danao. On the parametric linear complementarity problem. *Journal of Optimization Theory and Applications*, 95(2):445–454, 1997.

[25] J. A. De Loera, R. Hemmecke, and M. Köppe. Pareto optima of multicriteria integer linear programs. *INFORMS Journal on Computing*, 21(1):39–48, 2009.

[26] W. Dent, R. Jagannathan, and M. Rao. Parametric linear programming: some special cases. *Naval Research Logistics Quarterly*, 20(4):725–728, 1973.

[27] M. Ehrgott. *Multicriteria Optimization*, volume 2. Springer, 2005.

[28] M. Ehrgott and X. Gandibleux. Bound sets for biobjective combinatorial optimization problems. *Computers & Operations Research*, 34(9):2674–2694, 2007.

[29] M. Ehrgott, D. Tenfelde-Podehl, and T. Stephan. A level set method for multiobjective combinatorial optimization: application to the quadratic assignment problem. *Pacific Journal of Optimization*, 2(3):521–544, 2006.

[30] J. Filar, K. Avrachenkov, and E. Altman. An asymptotic simplex method for parametric linear programming. In *Information, Decision and Control, 1999. IDC 99. Proceedings. 1999*, pages 427–432. IEEE, 1999.

[31] B. Finkelstein and L. Gumenok. Algorithm for solving a linear parametric program when the a-matrix depends upon a parameter. *Ekonomicko Matematiceskie Metody*, 13:342–347, 1977.

[32] B. Gailly, M. Installe, and Y. Smeers. A new resolution method for the parametric linear complementarity problem. *European Journal of Operational Research*, 128(3):639–646, 2001.

[33] G. Gamrath, T. Koch, A. Martin, M. Miltenberger, and D. Weninger. Progress in presolving for mixed integer programming. *Mathematical Programming Computation*, pages 1–32, 2015.

[34] A. Ghaffari-Hadigheh, O. Romanko, and T. Terlaky. Sensitivity analysis in convex quadratic optimization: simultaneous perturbation of the objective and right-hand-side vectors. *Algorithmic Operations Research*, 2(2):94, 2007.

[35] A. Ghaffari-Hadigheh, O. Romanko, and T. Terlaky. Bi-parametric convex quadratic optimization. *Optimization Methods & Software*, 25(2):229–245, 2010.

[36] C. Goh and X. Yang. Analytic efficient solution set for multi-criteria quadratic programs. *European Journal of Operational Research*, 92(1):166–181, 1996.

[37] C. Goh and X. Yang. Analytic efficient solution set for bi-criteria quadratic network programs. *Nonlinear Analysis-Theory Methods & Applications*, 30(7):4309–4316, 1997.

[38] S. L. Graham, P. B. Kessler, and M. K. Mckusick. Gprof: A call graph execution profiler. *SIGPLAN Notices*, 17(6):120–126, June 1982.

[39] A. Grancharova and T. A. Johansen. *Explicit Nonlinear Model Predictive Control.* Springer, 2012.

[40] A. Gupta, S. Bhartiya, and P. Nataraj. A novel approach to multiparametric quadratic programming. *Automatica*, 47(9):2112–2117, 2011.

[41] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari. Multi-parametric toolbox 3.0. In *Control Conference (ECC), 2013 European*, pages 502–510. IEEE, 2013.

[42] M. Herceg, C. N. Jones, M. Kvasnica, and M. Morari. Enumeration-based approach to solving parametric linear complementarity problems. *Automatica*, 62:243–248, 2015.

[43] M. Hirschberger, R. E. Steuer, S. Utz, M. Wimmer, and Y. Qi. Computing the nondominated surface in tri-criterion portfolio selection. *Operations Research*, 61(1):169–183, 2013.

[44] M. Hirschberger, Y. Qi, and R. E. Steuer. Large-scale MV efficient frontier computation via a procedure of parametric quadratic programming. *European Journal of Operational Research*, 204(3):581–588, 2010.

[45] IBM-ILOG CPLEX 12.6 User's Manual. IBM, 2014.

[46] C. N. Jones and M. Morari. Multiparametric linear complementarity problems. In *Decision and Control, 2006 45th IEEE Conference on*, pages 5687–5692. IEEE, 2006.

[47] P. Jonker, G. Still, and F. Twilt. One-parametric linear-quadratic optimization problems. *Annals of Operations Research*, 101(1-4):221–253, 2001.

[48] N. Jozefowiez, G. Laporte, and F. Semet. A generic branch-and-cut algorithm for multiobjective optimization problems: Application to the multilabel traveling salesman problem. *INFORMS Journal on Computing*, 24(4):554–564, 2012.

[49] S. Kalajdzievski. *An Illustrated Introduction to Topology and Homotopy.* CRC Press, 2015.

[50] V. Kariwala and Y. Cao. Multiobjective control structure design: a branch and bound approach. *Industrial & Engineering Chemistry Research*, 51(17):6064–6070, 2012.

[51] R. Khalilpour and I. Karimi. Parametric optimization with uncertainty on the left hand side of linear programs. *Computers & Chemical Engineering*, 60:31–40, 2014.

[52] C. Kim. Parameterizing an activity vector in linear programming. *Operations Research*, 19 (7):1632–1646, 1971.

[53] G. Kiziltan and E. Yucaoğlu. An algorithm for multiobjective zero-one linear programming. *Management Science*, 29(12):1444–1453, 1983.

[54] T. Koch, T. Achterberg, E. Andersen, O. Bastert, T. Berthold, R. E. Bixby, E. Danna, G. Gamrath, A. M. Gleixner, S. Heinz, A. Lodi, H. Mittelmann, T. Ralphs, D. Salvagnin, D. E. Steffy, and K. Wolter. MIPLIB 2010. *Mathematical Programming Computation*, 3(2):103–163, 2011.

[55] M. Kostreva. Generalization of Murty's direct algorithm to linear and convex quadratic programming. *Journal of Optimization Theory and Applications*, 62(1):63–76, 1989.

[56] C. E. Lemke. Bimatrix equilibrium points and mathematical programming. *Management Science*, 11(7):681–689, 1965.

[57] Z. Li and M. G. Ierapetritou. A method for solving the general parametric linear complementarity problem. *Annals of Operations Research*, 181(1):485–501, 2010.

[58] G. Maier. A matrix structural theory of piecewise linear elastoplasticity with interacting yield planes. *Meccanica*, 5(1):54–66, 1970.

[59] I. Maros and C. Mészáros. A collection of QP instances. URL `http://www.doc.ic.ac.uk/~im/#DATA`.

[60] A. Martin. General mixed integer programming: Computational issues for branch-and-cut algorithms. In M. Junger and D. Naddef, editors, *Computational Combinatorial Optimization*, volume 2241 of *Lecture Notes in Computer Science*, pages 1–25. Springer, 2001.

[61] MATLAB R2013b. The MathWorks Inc., Natick, Massachusetts, 2013.

[62] MATLAB R2016a. The MathWorks Inc., Natick, Massachusetts, 2016.

[63] G. Mavrotas and D. Diakoulaki. Multi-criteria branch and bound: A vector maximization algorithm for mixed 0-1 multiple objective linear programming. *Applied Mathematics and Computation*, 171(1):53–71, 2005.

[64] K. Murty. On the parametric linear complementarity problem. *Engineering Summer Conference Notes, University of Michigan Ann Arbor, MI*, 1971.

[65] K. G. Murty and F.-T. Yu. *Linear Complementarity, Linear and Nonlinear Programming (Internet Edition)*. 1997.

[66] Y. Ohsawa. A geometrical solution for quadratic bicriteria location models. *European Journal of Operational Research*, 114(2):380–388, 1999.

[67] M. H. Overmars and J. Van Leeuwen. Dynamic multi-dimensional data structures based on quad-and kd trees. *Acta Informatica*, 17(3):267–285, 1982.

[68] Ö. Özpeynirci and M. Köksalan. An exact algorithm for finding extreme supported nondominated points of multiobjective mixed integer programs. *Management Science*, 56(12): 2302–2315, 2010.

[69] J.-S. Pang. A parametric linear complementarity technique for optimal portfolio selection with a risk-free asset. *Operations Research*, 28(4):927–941, 1980.

[70] J.-S. Pang, I. Kaneko, and W. Hallman. On the solution of some (parametric) linear complementarity problems with applications to portfolio selection, structural engineering and actuarial graduation. *Mathematical Programming*, 16(1):325–347, 1979.

[71] S. N. Parragh and F. Tricoire. Branch-and-bound for bi-objective integer programming. *Optimization Online*, 2015.

[72] P. Patrinos and H. Sarimveis. A new algorithm for solving convex parametric quadratic programs based on graphical derivatives of solution mappings. *Automatica*, 46(9):1405–1418, 2010.

[73] Y. Peng, G. Kou, Y. Shi, and Z. Chen. A multi-criteria convex quadratic programming model for credit data analysis. *Decision Support Systems*, 44(4):1016–1030, 2008.

[74] E. N. Pistikopoulos, V. Dua, N. A. Bozinis, A. Bemporad, and M. Morari. On-line optimization via off-line parametric optimization tools. *Computers & Chemical Engineering*, 26(2):175–185, 2002.

[75] E. N. Pistikopoulos, L. Dominguez, C. Panos, K. Kouramas, and A. Chinchuluun. Theoretical and algorithmic advances in multi-parametric programming and control. *Computational Management Science*, 9(2):183–203, 2012.

[76] A. Ponsich, A. Jaimes, and C. Coello. A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications. *Evolutionary Computation, IEEE Transactions on*, 17(3):321–344, 2013.

[77] A. Przybylski, X. Gandibleux, and M. Ehrgott. A two phase method for multi-objective integer programming and its application to the assignment problem with three objectives. *Discrete Optimization*, 7(3):149 – 165, 2010.

[78] A. Przybylski, X. Gandibleux, and M. Ehrgott. A two phase method for multi-objective integer programming and its application to the assignment problem with three objectives. *Discrete Optimization*, 7(3):149–165, 2010.

[79] QPlib2014. URL `http://www.lamsade.dauphine.fr/QPlib2014/doku.php`.

[80] T. K. Ralphs, M. J. Saltzman, and M. M. Wiecek. An improved algorithm for solving biobjective integer programs. *Annals of Operations Research*, 147(1):43–70, 2006.

[81] D.-M. K. Ritter. Ein verfahren zur lösung parameterabhängiger, nichtlinearer maximum-probleme. *Unternehmensforschung*, 6(4):149–166, 1962.

[82] O. Romanko. Parametric and multiobjective optimization with applications in finance. *ETD Collection for McMaster University*, 2010.

[83] H. Samet. *Foundations of Multidimensional and Metric Data Structures.* Morgan Kaufmann, 2006.

[84] K. Smimou. International portfolio choice and political instability risk: A multi-objective approach. *European Journal of Operational Research*, 234(2):546–560, 2014.

[85] F. Sourd and O. Spanjaard. A multiobjective branch-and-bound framework: Application to the biobjective spanning tree problem. *INFORMS Journal on Computing*, 20(3):472–484, 2008.

[86] B. Soylu. Heuristic approaches for biobjective mixed 0–1 integer linear programming problems. *European Journal of Operational Research*, 245(3):690–703, 2015.

[87] J. Spjøtvold, E. C. Kerrigan, C. N. Jones, P. Tøndel, and T. A. Johansen. On the facet-to-facet property of solutions to convex parametric quadratic programs. *Automatica*, 42(12): 2209–2214, 2006.

[88] J. Spjøtvold, P. Tøndel, and T. Johansen. Continuous selection and unique polyhedral representation of solutions to convex parametric quadratic programs. *Journal of Optimization Theory and Applications*, 134(2):177–189, 2007.

[89] R. E. Steuer, Y. Qi, and M. Hirschberger. Comparative issues in large-scale mean–variance efficient frontier computation. *Decision Support Systems*, 51(2):250–255, 2011.

[90] T. Stidsen, K. A. Andersen, and B. Dammann. A branch and bound algorithm for a class of biobjective mixed integer programs. *Management Science*, 60(4):1009–1032, 2014.

[91] M. Sun. A primogenitary linked quad tree data structure and its application to discrete multiple criteria optimization. *Annals of Operations Research*, 147(1):87–107, 2006.

[92] M. Sun and R. E. Steuer. Quad-trees and linear lists for identifying nondominated criterion vectors. *INFORMS Journal on Computing*, 8(4):367–375, 1996.

[93] K. Tammer. Parametric linear complementarity problems. *Lecture Notes in Pure and Applied Mathematics*, pages 399–418, 1997.

[94] P. Tøndel, T. A. Johansen, and A. Bemporad. An algorithm for multi-parametric quadratic programming and explicit mpc solutions. *Automatica*, 39(3):489–497, 2003.

[95] P. Tøndel, T. A. Johansen, and A. Bemporad. Further results on multiparametric quadratic programming. In *Proc. 42nd IEEE Conf. on Decision and Control*, volume 3, pages 3173–3178. Citeseer, 2003.

[96] H. Väliaho. A procedure for one-parametric linear programming. *BIT Numerical Mathematics*, 19(2):256–269, 1979.

[97] H. Väliaho. A unified approach to one-parametric general quadratic programming. *Mathematical Programming*, 33(3):318–338, 1985.

[98] H. Väliaho. A procedure for the one-parametric linear complementarity problem. *Optimization*, 29(3):235–256, 1994.

[99] H. Väliaho. Criteria for sufficient matrices. *Linear Algebra and Its Applications*, 233:109–129, 1996.

[100] H. Väliaho. Determining the handicap of a sufficient matrix. *Linear Algebra and its Applications*, 253(1):279–298, 1997.

[101] T. Vincent, F. Seipp, S. Ruzika, A. Przybylski, and X. Gandibleux. Multiple objective branch and bound for mixed 0-1 linear programming: Corrections and improvements for the biobjective case. *Computers & Operations Research*, 40(1):498–509, 2013.

[102] L. B. Willner. On parametric linear programming. *SIAM Journal on Applied Mathematics*, 15(5):1253–1257, 1967.

[103] B. Xiao. The linear complementarity-problem with a parametric input. *European Journal of Operational Research*, 81(2):420–429, 1995.

[104] J.-R. Yu and W.-Y. Lee. Portfolio rebalancing model using multiple criteria. *European Journal of Operational Research*, 209(2):166–175, 2011.

[105] C. Zopounidis, E. Galariotis, M. Doumpos, S. Sarri, and K. Andriosopoulos. Multiple criteria decision aiding for finance: An updated bibliographic survey. *European Journal of Operational Research*, 247(2):339–348, 2015.