Global Optimization of Fractional Programs with Applications to Engineering and Management Problems LIU JIANING 2017

Global Optimization of Fractional Programs with Applications to Engineering and Management Problems

LIU JIANING

SCHOOL OF MECHANICAL AND AEROSPACE ENGINEERING

2017

Global Optimization of Fractional Programs with Applications to Engineering and Management Problems

LIU JIANING

School of Mechanical and Aerospace Engineering

Manufacturing & Industrial Engineering Cluster

A thesis submitted to the Nanyang Technological University

in partial fulfillment of the requirement for the degree of

Doctor of Philosophy

2017

Acknowledgements

It is impossible for me to complete my thesis without the help of many people.

Above all, my sincere gratitude goes to my my supervisor, Dr. Jitamitra Desai for his consistently patient guidance and invaluable supervision and efforts of my Ph.D. study.

I am grateful to my classmate Qi Xiaofei for her help to improve my skills in MATLAB coding. I am also grateful to the other members of our research group. It is really a pleasure to collaborate with Wang Kai, Rakesh Prakash and Lian Guan.

I would also like to extend my gratitude to the professors and staffs of the Division of System and Enginnering Management of MAE for the courses, seminars, and for their guidance and support. I would like to thank the University for providing excellent working conditions and the Scholarship which supported me to undertake this research and complete my thesis.

I am also indebted to my friends from NTU: Zhang He, Ji Chenzhen, and An Jinliang and so on. Thank you for making my Ph.D. life colorful and fun.

I would also like to thank my family, since I obtained strength and confidence from their encouragement, support.

Finally, my beloved wife, Jin Jin, thank you for your eternal love, persistent support and endless patience.

Contents

Abstract

In this dissertation, we focus on deriving globally optimal algorithms for three specialized fractional programming problems arising in the context of the independent set problem and management applications in wireless communications. Most of the research efforts thus far involving fractional programs have focused on solving various classes of the single-ratio fractional programs, or multi-ratio cases involving linear fractional functions. However, fractional programming problems become substantially more difficult as the number of ratios in the objective function increases, and this complexity is further amplified when the involved terms are higher-order polynomials. While some classes of multiple-ratio problems can be solved to optimality by recasting them as equivalent *0-1 mixed-integer programming problems*, and embedding them within a tailored branch-and-bound algorithm, there is yet significant scope to devise specialized algorithms for solving the generic class of sum of ratios fractional programs, particularly for the case where the involved functions are higher-degree polynomials.

We begin by investigating continuous optimization approaches, notably fractional programming methods, to determine the stability number (or independence number) of a graph. Traditionally, this problem has been solved using integer programming methods, but these methods have been known to suffer from a number of shortcomings, in addition to the complexity of dealing with discrete variables. In this context, a new class of *vertex sets* is defined, and the structure of these vertex sets is utilized to derive explicit characterizations of the number of alternate optima present in both discrete and continuous formulations. Moreover, these vertex sets also enable a simple, yet powerful, construction procedure to efficiently determine maximal independent sets. We also develop a global optimization algorithm to solve the FP formulation, and we demonstrate that this continuous approach stays on par with the 0-1 discrete formulation with respect to various performance metrics. As seen in our numerical experiments, we showed that the computational time required per optimal solution is comparable and in some instances lower for Problem FP as compared to Problem MIS (as the number of alternate optima for Problem FP is significantly greater when

compared to Problem MIS)

Having established the importance of fractional programming formulations as a viable source for solving hard optimization problems, in the next phase of this research, we focus on solving real-world applications arising in the context of cellular network design. We present a set of (exact and approximate) mathematical models and algorithms for determining the set of (globally) optimal distributed antenna deployments and the supported user demand in cellular code division multiple access (CDMA) systems. We focus on the uplink (user-to-base) formulation and assume that the base station combines all the received signals at each of the antennas using path-gain based weights. In CDMA systems, as all users occupy the system bandwidth at the same time thereby interfering with each other, this results in complicated mixed-integer 0-1 multi-linear programming problem, where the objective function maximizes the total system capacity, while ensuring that the minimum signal-to-interference plus noise ratio (SINR) constraints and maximum transit power constraints for each user are satisfied. This highly nonlinear, nonconvex problem is reformulated to yield a tight mixed-integer 0-1 linear programming representation via the addition of several auxiliary variables and constraints, and a specialized algorithm is designed to determine globally optimal solutions.

Finally, we extend the above developed methodology for optimally locating DAS antennas in CDMA cellular networks in the presence of femtocells. Although a femtocell is a simple plug-and-play device, its location requires pre-approval by the service provider. The introduction of femtocells into a cellular network can be very beneficial to both service providers as well as end users. Once again, our focus is on developing an analytical framework that provides a way of computing optimal DAS deployments so that the total capacity is maximized in the CDMA cellular network. Such a framework would be critical for service providers who are interested in adopting the use of femtocells.

In conclusion, this thesis provides an algorithmic framework for deriving globally optimal solutions for the class of fractional programming problems, and this approach is validated by solving various practical problems arising in engineering and management applications.

List of Figures

List of Tables

List of Symbols

$\mathbb{R}$	Real number
$\top$	Transpose
$G=(V,\ E)$	A simple, undirected graph
V	Vertex set of G
E	Edge set of G
$G(W)$	Subgraph induced by W on G
$N(i)$	Set of neighbors of vertex i
$d_i \equiv \lvert N(i)\rvert$	Degree (or cardinality) of vertex $i \in V$
$\bar{d} = \sum_{i=1}^{n} d_i \Big/ n$	The average degree of all vertices in G
$\Delta = \Delta(G) = \max_{i\in V}\{d_i\}$	Maximum degree among all vertices of G
$\bar{G}=(V,\ \bar{E})$	Complement graph of G
A_G	Adjacency matrix of G
$\alpha(G)$	Independence number of G
$v[\bullet]$	The optimum objective function value of a Problem $[\bullet]$
$\overline{\text{MIS}}$	Linear programming (**LP**) relaxation of MIS
$\overline{\text{RFP}}$	Linear programming relaxation of Problem RFP
$\omega(G)$	Clique number, the cardinality of a maximum clique in G
I	An independent set
$C^{(k)}$	Vertex set of $k \in I$
$\tilde{C}^{(k)}$	Refined (disjoint) vertex sets
Λ_{MIS}	The number of alternate optimal solutions to Problem MIS

Λ_{FP}	The number of alternate optimal solutions to Problem FP
$\alpha^{LP}(G)$	The optimum objective function value of Problem $\overline{\mathrm{RFP}}$
UB	Upper bound
LB	Lower bound
ρ	Graph density
$\Re$	A CDMA network region
$(\alpha_i, \beta_i) \in \Re$	The location of demand node i
$\tilde{T}_i$	Path gains from demand node i to the given base
$T_i(x, y)$	Path gains from demand node i to a DAS antenna located at position (x, y)
$\sqrt{T_{0i}} e^{j\phi_{0i}}$	Signal gain between each user at demand node i and the given base
$\sqrt{T_{ki}} e^{j\phi_{ki}}$	Signal between this user and an antenna deployed at location $(x_k, y_k), \quad k = 1,...K$
ϕ_{ki}	Phases of the signal gains from user i to a DAS antenna at location k
ω_{ki}	Weight of the output streams at a DAS antenna at location k
P_i	Transmit power of user i
W	System bandwidth,
R	Supported rate of each admitted user,
ηW	Power of the AWGN at the receiver
SINR	Signal-to-Interference plus Noise Ratio
Γ	a minimum SINR
χ	A Gaussian random variable with parameters (μ, σ^2)

Chapter 1

Introduction

Fractional Programming (**FP**) is a branch of nonlinear optimization, in which the objective function contains one or several ratios, and such optimization problems are called *fractional problems* or *hyperbolic programs*. Neumann (1937) first introduced the notion of fractional programming, where an equilibrium model that determines the growth rate of an economy is represented as the maximum of the smallest of several output/input ratios. Another early work appears in Isbell and Marlow (1956), who solved a constrained ratio optimization problem (essentially a fractional program), and since then, there has been sustained interest in the field of fractional programming. However, a systematic study of fractional programming only began with the work of Charnes and Cooper (1962), who published their classical paper "Programming with Linear Fractional Functionals", and the now well-established terminology "fractional program" was first advocated in this work.

1.1. Single Ratio Fractional Programs

Let f, g, h_k, $k=1,...m$, denote real-valued functions, which are defined on a set $C \in \mathbb{R}^n$. Consider,

$$q(x) = \frac{f(x)}{g(x)} \tag{1.1}$$

over the domain set $S \equiv \{x \in C:\ h_k(x) \le 0,\ \forall k = 1,...,m\}$, assuming that $g(x) > 0$ on C. Then, the nonlinear program

$$(P) \qquad \sup\{q(x):\ \ x \in S\} \tag{1.2}$$

is called a (*single-ratio*) *fractional program*.

If f, g, and h_k are *affine* functions and $C \in \mathbb{R}^n_+$ (the nonnegative orthant of $\mathbb{R}^n$), then (P) is called a *linear fractional program*, with the following expression:

$$\sup\left\{\frac{c^\top x+\alpha}{d^\top x+\beta} : Ax \le b,\ x \ge 0\right\}, \tag{1.3}$$

where $c, d \in \mathbb{R}^n$, $\alpha, \beta \in \mathbb{R}$, the superscript '$\top$' denotes the transpose, $A \in \mathbb{R}^{m\times n}$ is a real-valued $(m \times n)$ matrix, and $b \in \mathbb{R}^m$. The fractional program (P) is called a *quadratic fractional program* if $C \in \mathbb{R}^n_+$, f and g are quadratic functions defined everywhere on the domain, and the h_k's are affine. Moreover, (P) is said to be a *concave* fractional program, if the numerator f is concave on *C*, and *g*, h_k are convex on *C*, where *C* is a convex set. Moreover, it is assumed that f is nonnegative on S if g is not affine.

Single-ratio fractional programs dominated the literature almost exclusively until the early 1980s. Many of these results are compiled in the first monograph on fractional programming by Schaible (1978). This article covers practical applications, theoretical results, and algorithms for single-ratio fractional programs, and it became a basis for Schaible (1981), Schaible and Ibaraki (1983), and for surveys by others. A more recent, detailed survey of single-ratio fractional programming applications can be found in Stancu-Minasian (1997).

There have been a plethora of algorithms for solving linear or more generally concave single-ratio fractional programs, namely (P), which are briefly reviewed below.

Solving Problem (P) directly

Concave (linear) fractional programs have some common properties with concave (linear) programs, because of the following properties of generalized concavity (and in addition, generalized convexity in the linear case) of the objective function $q(x) = f(x)/g(x)$ (see Avriel *et al.*, 1988 and Martos, 1975):

(i) A local maximum is a global maximum;

(ii) A maximum is unique if either the numerator is strictly concave or the denominator is strictly convex;

(iii) A solution of the Karush-Kuhn-Tucker (KKT) optimality conditions is a maximum, assuming f, g, h_k are differentiable on the open set C;

(iv) A maximum is obtained at an extreme point of the convex polyhedron S of a linear fractional program (provided an optimal solution exists).

Due to properties (i) and (iii), single ratio concave fractional programs can be solved using standard concave programming algorithms. Moreover, certain concave programming methods can also be applied when the objective function is *quasiconcave*, such as the Frank–Wolfe linearization method (Craven, 1988 and Martos, 1975). A cutting plane method was also developed by Boncompte and Martinez-Legaz (1991) for concave fractional programs, based on the upper subdifferentiability of the objective function. If (P) is a linear fractional program, then property (iv) can be used to calculate a maximum $\bar{x}$ by determining a finite sequence of extreme points x_i of S with increasing values $q(x_i)$ converging to $\bar{x}$. Thus, a simplex-like procedure can be used to solve linear fractional programs (Martos, 1975).

Solving an Equivalent Problem (P_{eq})

While concave programming algorithms cannot be directly deployed to solve concave fractional programs (Martos, 1975), a strategy that transforms concave fractional programs into a concave program can be suitably utilized in such cases. Towards this end, define:

$$y = \frac{x}{g(x)} \quad \text{and} \quad t = \frac{1}{g(x)}, \tag{1.4}$$

and this transformation reduces (P) to an equivalent problem given by

$$P_{eq}: \qquad \sup\left\{ tf\left(\frac{y}{t}\right) : (y,\ t) \in \hat{S} \right\}, \tag{1.5}$$

where the region

$$\hat{S} \equiv \left\{ th_k\left(\frac{y}{t}\right) \le 0,\ tg\left(\frac{y}{t}\right) \le 1, \quad \frac{y}{t} \in C,\ t > 0 \right\}. \tag{1.6}$$

If $(\bar{y},\ \bar{t})$ is an optimal solution of (P_{eq}), then $\bar{x} = \dfrac{\bar{y}}{\bar{t}}$ is an optimal solution of (P). Solving (P_{eq}) rather than (P) may be particularly appropriate when the numerator f and the denominator g possess a certain algebraic structure. For example, the maximum probability model or certain portfolio selection models have an affine numerator and a denominator which is the square root of a convex quadratic form. In this case (P_{eq}) reduces to a concave quadratic program, and hence (P) can directly be solved by one of the standard quadratic programming techniques (Schaible, 1983).

In the special case of *linear* fractional programs, i.e., $\sup\left\{\frac{c^\top x+\alpha}{d^\top x+\beta}:\ \ Ax\le b,\ x\ge 0\right\}$, transformation (1.4) given in Charnes and Cooper (1962) yields the linear program:

$$\sup\left\{c^\top y+\alpha t:\ \ (y,\ t)\in\hat{S}\right\}, \tag{1.7}$$

with the feasible region $\hat{S}$ being represented by the relations

$$\hat{S}\equiv\left\{Ay-bt\ge 0,\quad d^\top y+\beta t=1,\ y\ge 0,\ t>0\right\}. \tag{1.8}$$

Thus, the problem $\sup\left\{\frac{c^\top x+\alpha}{d^\top x+\beta}:\ \ Ax\le b,\ x\ge 0\right\}$ can now be solved by the simplex method, and for a comparison with other linear fractional programming methods, we refer the reader to (Schaible, 1983).

Swarup (1965) introduced an alternative approach for solving linear fractional functional programs, where beginning with a basic feasible solution and conditions for optimality criteria, the underlying method works like the simplex technique resulting in an improving sequence of solutions until the optimum is achieved.

Solving a Parametric Problem (P_q)

There is a rich class of algorithms for solving the single-ratio fractional program based on the following parametric problem associated with (P):

$$(P_q)\qquad \max\left\{f(x)-qg(x):\ \ x\in S\right\}, \tag{1.9}$$

where, $q\in\mathbb{R}$ is a parameter. The program (P_q) is sometimes numerically more tractable than the program (P) (for details, see Sniedovich, 1988). In the following, we assume that S is compact and f,g are continuous on S. Let $F(q)$ denote the optimal value of the objective function of (P_q). Then, $F(q)$ is a strictly decreasing, convex function which has a root at $q=\bar{q}$. An optimal solution $\bar{x}$ of ($P_{\bar{q}}$) is also an optimal solution of (P) with $\bar{q}=\frac{f(\bar{x})}{g(\bar{x})}$, and thus solving ($P$) is equivalent to finding the unique root of the nonlinear equation $F(q)=0$. Given these favorable properties of $F(q)$, Ibaraki (1983) applied interval-type algorithms that generate a succession of intervals with decreasing amplitude containing $q=\bar{q}$, to calculate the root of the function at $q=\bar{q}$.

The most well-known application of Newton's method for solving the parametric problem is the Dinkelbach's algorithm (Dinkelbach, 1967). A more refined version of Dinkelbach's method was suggested Schaible (1976) and was further improved in Pardalos and Phillips (1991). Dinkelbach's original iterative algorithm is based on the following iterative scheme:

Dinkelbach(S, f, g)**:**

1. Select some $x^{(0)} \in S$. Set $\lambda^{(0)} := f\left(x^{(0)}\right) \big/ g\left(x^{(0)}\right)$ and $k := 0$.
2. Solve the constrained global optimization problem

   $$\max_{x\in S}\left(f(x)-\lambda^{(k)}g(x)\right)$$

   to get the optimal solution point $x^{(k+1)}$.
3. If $f\left(x^{(k+1)}\right)-\lambda^{(k)}g\left(x^{(k+1)}\right)=0$, then set $x^{*} := x^{(k+1)}$ and $\lambda^{*} := \lambda^{(k+1)}$

Extending (P), consider the following *generalized fractional programming* problem given by: $\bar{\lambda}=\inf_{x\in X}\left\{\max_{1\le i\le m}\left\{f_i(x)\big/ g_i(x)\right\}\right\}$, where X is a nonempty subset of $\mathbb{R}^n$, f_i and g_i are continuous on an open set $\tilde{X}$ in $\mathbb{R}^n$, including the $\mathrm{cl}(X)$ (the *closure* of X), and $g_i(x)>0$ for all $x\in\tilde{X}$, $1\le i\le m$. When $m=1$, then the generalized fractional program reduces to the classical fractional programming problem (P). This type of problem occurs frequently when we optimize some kind of efficiency measures expressed as a ratio. Algorithms for such problems, reviewed in Schaible (1983, 1995) and Schaible and Ibaraki (1983), can be seen as generalizations of the approach proposed by Dinkelbach (1967) for the case $m = 1$ to determine the root of the equation $F(\lambda) = 0$, where $F(\lambda)$ is the optimal value of the parametric program:

$$(P_\lambda) \qquad F(\lambda) = \inf_{x \in X} \left\{ \max_{1 \le i \le m} \{ f_i(x) - \lambda g_i(x) \} \right\}. \tag{1.10}$$

1.2. Maximizing a Sum of Ratios

Consider the sum of ratios fractional program:

$$\sup \left\{ \sum_{i=1}^{p} q_i(x) : \ x \in S \right\}, \tag{1.11}$$

where $q_i(x) = f_i(x) / g_i(x)$ and $g_i(x) > 0$.

Problem (1.11) arises naturally in decision making when several rates are to be optimized simultaneously and a compromise is sought which optimizes a weighted sum of these rates, and it also includes the case when some ratios are not proper quotients, *i.e.*, $g_i(x) = 1$. This describes situations where a compromise is sought between absolute and relative terms like profit and return on investment (profit/capital) or return and return/risk. Additional applications of (1.11) are surveyed in Schaible (1996).

An important class of 0-1 optimization problems is the maximization (or minimization) of a *sum of ratios of linear 0-1 functions*:

$$\text{Maximize} \qquad f(x) = \sum_{j=1}^{m} \frac{a_{j0} + a_j^\top x}{b_{j0} + b_j^\top x} \tag{1.12a}$$

$$\text{Subject to:} \quad Dx \le c, \tag{1.12b}$$

$$x \in \{0, 1\}^n, \tag{1.12c}$$

where $a_j \in \mathbb{R}^n$, $b_j \in \mathbb{R}^n$, $a_{j0} \in \mathbb{R}$, $b_{j0} \in \mathbb{R}$, $D \in \mathbb{R}^{k \times n}$ and $c \in \mathbb{R}^k$.

Problem (1.12a)–(1.12c) is referred to as *0-1 linear fractional programming problem,* and note that, for some x in the feasible region, if the term $b_{j0} + b_j^\top x$ is equal to zero, then

Problem (1.12a) - (1.12c) may not have a finite optimum. Therefore, we assume that

$$b_{j0} + b_j^\top x \neq 0, \text{ for all } x \in \{0,\ 1\}^n, \text{ and } j = 1,\ldots m. \tag{1.13}$$

Furthermore, a stricter assumption is imposed, which requires that all denominators in (1.12a) are positive, *i. e.*,

$$b_{j0} + b_j^\top x > 0, \text{ for all } x \in \{0,\ 1\}^n, \text{ and } j = 1,\ldots m. \tag{1.14}$$

Li (1994) and Wu (1997) suggested a straightforward linearization technique for Problem (1.12) based on a simple well-known idea: A polynomial mixed 0-1 term $z = xy$, where x is a 0-1 variable and y is a continuous, strictly positive variable, can be represented by the following linear inequalities: (1) $y - z \le K - Kx$; (2) $z \le y$; (3) $z \le Kx$; (4) $z \ge 0$, where K is an upper bound on y.

Assuming that $z \ge 0$ is satisfied, define a new variable y_j for each ratio in problem (1.12), i.e.,

$$y_i = \frac{1}{b_{j0} + \sum_{i=1}^{n} b_{j\,i} x}. \tag{1.15}$$

Then, the fractional 0-1 programming problem (1.12) can be equivalently expressed as:

$$\max_{x \in \{0,1\}^n} \sum_{j=1}^{m} a_{j0}\, y_j + \sum_{j=1}^{m} \sum_{i=1}^{n} a_{j\,i} u \tag{1.16a}$$

$$\text{subject to:} \quad Dx \le c \tag{1.16b}$$

$$b_{j0} y_j + \sum_{i=1}^{n} b_{ji} u_{ji} = 1, \ j = 1,\ldots m \tag{1.16c}$$

$$y_j - K_j (1 - x_i) \le u_{ji} \le K_j x_i, \quad j = 1,\ldots m; \ \ i = 1,\ldots n \tag{1.16d}$$

$$0 \le u_{ji} \le y_j, \quad j = 1,\ldots m; \ \ i = 1,\ldots n, \tag{1.16e}$$

where a new variable u_{ji} is introduced for each nonlinear term $y_j x_i$, and K_j is an upper bound on y_j. Additionally, though similar in spirit to (1.16), linear mixed 0-1 integer reformulations as well as other related issues are carefully discussed in Tawarmalani (2002).

As noted in the above discussion, most of the research efforts thus far have been focused on solving various classes of the single-ratio fractional program (*P*). However, fractional programming problems become substantially more difficult as the number of ratios in the objective function increases. While some classes of multiple-ratio problems can be solved to optimality by recasting them as equivalent *0-1 mixed-integer programming problems*, and embedding them within a tailored branch-and-bound algorithm (see Li, 1994, Tawarmalani *et al*., 2002, and Wu, 1997), there is yet significant scope to devise specialized algorithms for solving the generic class of sum of ratios fractional programs, particularly for the case where the involved functions are higher-degree polynomials.

In this dissertation, we focus on deriving globally optimal algorithms for three specialized fractional programming problems arising in the context of the independent set problem and management applications in wireless communications. Each of these tasks is summarized below.

1.3. Research Objectives

As aforementioned, our research primarily focuses on three scenarios: (i) A fractional programming approach to determine the stability number of a graph; (ii) A discrete optimization framework for distributed antenna system (DAS) locations in CDMA cellular networks; and (iii) Extending this DAS framework for CDMA networks in the presence of an emerging technology called femtocells.

Each of these research objectives is briefly described below.

Research Objective 1: *A fractional programming approach for the stability number*

Let $G=(V,\ E)$ be a simple, undirected graph with vertex set $V \equiv \{1,\ldots,n\}$ and edge set $E \subseteq V \times V$. For any subset $W \subseteq V$, let $G(W)$ denote the subgraph induced by W on G. A set of vertices $I \subseteq V$ is called an *independent set* (or *stable set*) if for every $i,\ j \in I$, $(i,\ j) \notin E$, *i.e.*, the graph $G(I)$ induced by I is edgeless. An independent set is *maximal* if it is not a subset of any larger independent set, and *maximum* if there are no larger independent sets in the graph. The cardinality of a maximum independent set of G is called the *independence number* of G (also referred to as the *stability number*) and is denoted by $\alpha(G)$.

The maximum independent set problem has been proven to be NP-Hard, and thus finding a global optimum to this problem is a computationally challenging task. This has led to a plethora of integer programming formulations being developed for finding maximum independent sets, and thence the maximum independence number of a graph. Amongst the notable ones is the 0-1 linear programming formulation, where the maximum independent set, based on its definition above, is characterized by maximizing the number of nodes subject to a neighbor-node disconnectivity constraint. For a detailed literature survey on the maximum independent set problem, its complexity and applications, along with related discussion on exact algorithms and heuristic procedures, we refer the reader to Pardalos *et al.* (1999).

However, it is only recently that attempts have been made to solve the independence number problem via global optimization approaches. In this research, we focus on the following fractional programming formulation, given by:

$$FP: \max\left\{ f(x) : 0 \le x_i \le 1, \ \forall i \in V \right\}; \qquad f(x) = \sum_{i \in V} \frac{x_i}{1 + \sum_{j \in N(i)} x_j} \tag{1.17}$$

where $N(i)$ is the neighbor set of node i, and determine a globally optimal algorithm for solving this problem.

Moreover, when comparing the fractional program FP with the 0-1 MIP formulation for determining sets that is more traditionally used in the literature, we also derive tight lower bounds on the number of alternate optima present in both formulations, and this in turn explains the computational comparisons provided between these alternative approaches.

<u>Research Objective 2</u>: *A fractional optimization framework for DAS in cellular networks.*

Due to the rapid development of wireless technology, enabling not only voice but also data and multimedia options, cellular phones have become an indispensable part of our everyday life. But an often recurring complaint among consumers concerns the imperfect coverage of current cellular networks and this problem is particularly acute in indoor environments. In typical cellular networks, a hand-held terminal, referred to as a *user*, employs a single-antenna radio to communicate with a local *base station*. The base station comprises $L > 1$ sufficiently spaced *antenna elements* (AEs) that transmit and receive signals with a user located within its *coverage* (or *service*) area called a *cell*. Because wireless signals use multiple paths when propagating from a transmitter to a receiver, the signal received at an antenna is a superposition of multiple copies of the transmitted signal. This results in either constructive or destructive interference, which amplifies or attenuates the power of the received signal. Strong destructive interference, *i.e.*, a deep fade, is extremely undesirable as it results in a (temporary) failure of communication. Having $L > 1$ sufficiently spaced base station antennas results in L independent sets of fading paths between a transmitter and receiver, thereby reducing the overall probability of a deep fade from p (for a single transmit-receive antenna pair), where $0 < p < 1$, to the order of p^L. Thus, the use of L base station antennas helps protect transmissions made between the user and the base station against multipath fading, and this reliability improvement is referred to as the *diversity gain* of applying multiple antennas in wireless communications.

Distributed antenna systems **(DAS)** provide an alternate arrangement to achieve improved diversity gains in cellular networks by simultaneously increasing the overall capacity and coverage performance. In the DAS framework, the L antenna elements of a base station are deployed throughout the cell, as opposed to being installed at a common radio tower, and are connected to the base station using dedicated links, *e.g.*, fiber optic, coax, or radio connections. An immediate advantage of distributing antennas (over co-locating them at a tower) is improved coverage within the cell, and can even lead to larger cells. Furthermore, by shortening the distance between a user and the antenna tower (the *access distance*), distributed antenna systems lead to a significant reduction in the required transmit power, which in turn reduces co-channel interference and results in improved capacity. Moreover, these L geographically distributed antennas can provide the same reliability or diversity gains as L antennas located at a common base station tower, provided that the antenna elements are placed with sufficient inter-antenna spacing.

Several recent studies have demonstrated the aforementioned benefits of DAS deployment in cellular networks. To achieve optimal capacity gains using a DAS, the antenna locations must be carefully chosen such that the transmit powers are minimized, *i.e.*, interference is minimized and capacity is maximized. As demonstrated by Dai (2008), with uniformly distributed users and regularly spaced antennas within a cell, the sum transmit power of a DAS is less than 5% of that consumed by conventional co-located antennas. This tremendous power improvement results in part due to reduced access distance and because the chosen antenna locations closely "match" the user distribution. However, for more general, non-uniform cases, optimal antenna locations based on the geography of user demand are required to effectively harness the capacity gain of the DAS architecture.

Hence, an analytical framework that provides a way of computing optimal antenna deployments so that the total capacity is maximized in a DAS setting would be critical for service providers considering the DAS approach. To the best of our knowledge, no such framework currently exists. In this paper, we pose and solve the problem of determining optimal distributed antenna deployments for maximizing the total number of admitted users in a CDMA system.

Research Objective 3: *A fractional optimization framework for DAS in cellular networks with femtocells*.

An upcoming cellular network technology known as "femtocells" provides a promising option for cellular companies to improve their coverage, and provides high quality service in a cost effective manner. However, as with any newly developed technology, femtocells face a number of issues related to widespread adaptability that are yet to be resolved. Such problems include interference management, handoff mechanisms, security concerns, etc. For example, when an unauthorized cell phone is within the coverage of a femtocell, it would not be able to connect the femtocell and instead would rely on obtaining coverage directly from the distant macrocell base station. This user's mobile connectivity would experience significant negative interference from the femtocell. Thus, further research is required to understand the steps that need to be taken to alleviate interference from neighboring femtocells, and consequently improve indoor coverage. To solve the aforementioned problems, in this research, we develop an analytical framework for determining optimal antenna deployments so that the total capacity is maximized in CDMA systems in the presence of femtocells.

The underlying structure of the problems identified above conforms to that of sum of ratios fractional programming problems, thereby facilitating an application of convexification techniques for solving problems arising in these domains. Furthermore, the modeling and algorithmic developments in this dissertation can be incorporated within global optimization software to make them more robust and effective, thereby advancing the computational solvability of nonconvex programs.

1.4. Organization of the Thesis

The remainder of this thesis is structured as follows. In Chapter 2, we introduce a fractional programming formulation to determine the independence number of a graph and describe several mathematical properties related to this formulation. Next, the fractional program is linearized in a higher dimensional space by the addition of auxiliary variables and convex/concave outer-envelope constraints. This linearized formulation is then embedded within a specialized branch-and-bound algorithm to determine the independence number. Several computational results are illustrated to demonstrate the efficacy of the proposed approach. In Chapter 3, we begin by reviewing some key elements of CDMA networks and derive various exact and approximate mathematical formulations for locating distributed antenna systems to maximize the capacity of CDMA networks, and prescribe tailored algorithmic methodologies for determining globally optimal solutions for this problem. Then, Chapter 4 presents a detailed description of the DAS framework for CDMA networks in the presence of femtocells, and finally, Chapter 5 provides a summary and delineates extensions for future research.

Chapter 2

On Solving a Fractional Program to Determine Independent Sets

The maximum independent problem is a classical problem in combinatorial optimization with important applications in different domains. Due to its vital role in several theoretical fields and its applicability across a wide variety of practical settings, the *maximum independent set* (**MIS**) problem has been extensively studied in the literature.

In our research, we study the NP-Hard maximum independent set problem from a continuous fractional programming (FP) formulation perspective, and compare and contrast this FP formulation with the well-known 0-1 linear programming variant of this problem. In this context, a new class of *vertex sets* is defined, and the structure of these vertex sets is utilized to derive explicit characterizations of the number of alternate optima present in both discrete and continuous formulations. Moreover, these vertex sets also enable a simple, yet powerful, construction procedure to efficiently determine maximal independent sets. We also develop a global optimization algorithm to solve the FP formulation, notably fractional programming methods and we demonstrate that this continuous approach stays on par with the 0-1 discrete formulation with respect to various performance metrics. Traditionally, this problem has been solved using integer programming methods, but these methods have been known to suffer from a number of shortcomings, in addition to the complexity of dealing with discrete variables. In our approach, we consider a fractional programming approach, wherein we tradeoff the loss of linearity in the problem formulation by having to deal with only continuous variables. Some fundamental properties of fractional programming are listed, and proofs for some of them are given. As seen in our numerical experiments, we showed that the computational time required per optimal solution is comparable and in some instances lower for Problem FP as compared to Problem MIS (as the number of alternate optima for Problem FP is significantly greater when compared to Problem MIS).

The remainder of this chapter is structured as follows. In Section 2.1, we review some required notations and definitions of the graph theory. In Section 2.2, we begin by revisiting some mathematical properties of the fractional programming formulation, and several new results, notably the construction and associated properties of the newly defined vertex sets as well as characterizations of the number of alternate optima, are derived. Then, in Section 2.3, the fractional program is linearized in a higher dimensional space by the addition of suitably defined auxiliary variables and convex/concave outer-envelope constraints. This linearized formulation is then embedded within a specialized globally optimal branch-and-bound algorithm to determine the independence number. Next, in Section 2.4, some computational results and insights are presented to demonstrate the efficacy of the proposed approach, and finally, Section 2.5 summarizes the contributions of this paper and presents some pointers for future research.

2.1 Introduction

Let $G = (V,\ E)$ be a *simple, undirected* graph with vertex set $V \equiv \{1,\ldots,n\}$ and edge set $E \subseteq V \times V$. For any subset $W \subseteq V$, let $G(W)$ denote the subgraph *induced* by W on $G = (V,\ E)$. Now, let $N(i)$ denote the set of *neighbors* of vertex i, and let $d_i \equiv |N(i)|$ represent the *degree* (or *cardinality*) of vertex $i \in V$. We denote $\Delta = \Delta(G) = \max_{i \in V}\{d_i\}$, to be the m*aximum degree* among all vertices of G, and moreover, let $\bar{d} = \sum_{i=1}^{n} d_i \Big/ n$ represent the *average degree* of all vertices in G. Furthermore, let $\bar{G} = (V,\ \bar{E})$ denote the complement graph of G, where $\bar{E} = (V \times V) \setminus E$. Finally, let A_G we denote the *adjacency matrix* of G, where $A_G = \left[a_{ij}\right]_{(i,\ j) \in V \times V}$ is such that $a_{ij} = 1$ if $(i,\ j) \in E$ and $a_{ij} = 0$ otherwise.

DEFINITION 1. *A set of vertices* $I \subseteq V$ *is called an independent set (or stable set) if for every* $i,\ j \in I$, $(i,\ j) \notin E$*, i.e., the graph* $G(I)$ *induced by* I *is edgeless.*

An independent set is said to be *maximal* if it is not a subset of any larger independent set, and *maximum* if there are no larger independent sets in the graph. The cardinality of a maximum independent set of G is called the *independence number* of G (also referred to as the *stability number*), and is denoted by $\alpha(G)$. There are many practical applications for the maximum independent set problem, including but not limited to the ones found in signal transmission analysis, classification theory, economics, and social networks (see Asmussen, 1987; Cambini et al., 1990; Charnes et al., 1978; and Schaible, 1978). The maximum independent set problem has been proven to be NP-Hard (Garey and Johnson, 1979), and thus finding a global optimum to this problem is a computationally challenging task.

This has led to a plethora of *integer programming* (**IP**) formulations for finding a maximum independent set, and correspondingly the independence number of a graph. The most well-known approach is a *0-1 linear programming formulation*, where the objective function seeks to maximize the number of nodes included in a maximum independent set, subject to a neighbor-node disconnectivity constraint. This *maximum independent set* (**MIS**) formulation can be derived as follows. Define,

$$x_i = \begin{cases} 1, & \text{if node } i \text{ belongs to the maximum independent set} \\ 0, & \text{otherwise} \end{cases}. \tag{2.1a}$$

MIS: Maximize $$\sum_{i=1}^{n} x_i \tag{2.1b}$$

subject to: $$x_i + x_j \leq 1, \quad \forall (i, j) \in E, \tag{2.1c}$$

$$x_i \in \{0, 1\}, \quad \forall i = 1, \ldots, n. \tag{2.1d}$$

This formulation has been well-studied in the IP literature, and many approaches have been established for solving this problem. The optimal solution to MIS determines a maximum independent set and $\nu[\text{MIS}]$, where $\nu[\bullet]$ denotes the optimum objective function value of a Problem $[\bullet]$, and gives us the independence number of the graph. However, apart from the obvious difficulties of dealing with the presence of discrete variables, this formulation has been shown to suffer from a number of shortcomings. Specifically, it is well-known that an optimal solution to the *linear programming* (**LP**) relaxation of MIS, denoted as $\overline{\text{MIS}}$, satisfies the "half-integrality" property, *i.e.*, all the x- variables are $\left(0, \frac{1}{2}, 1\right)$-valued, and thus the strength and quality of the upper bound, *i.e.*, $v\left[\overline{\text{MIS}}\right]$, obtained by solving this LP relaxation is often very weak. To circumvent these difficulties and to strengthen the underlying LP relaxation, several classes of cutting planes, including facet-defining inequalities for the stable set polytope, have been developed (see Rebennack (2008) for further details).

REMARK 1. Note that a *feasible solution* to Problem $\overline{\text{MIS}}$ can be easily obtained by using the following procedure:

Table 1. A feasible solution to LP relaxation of Problem MIS

Algorithm 1. INPUT: $G=(V, E)$; OUTPUT: $\bar{x}$ feasible to LP relaxation of MIS
For $i=1,\ldots,n$, do:
If $d_i = 0$, set $\bar{x}_i = 1$;
Else, if $d_i \geq 1$, set $\bar{x}_i = 1/2$.
END

Figure (1a) illustrates an optimal solution to the LP relaxation of Problem MIS, where the contribution of the x-variables towards the objective function value has been indicated using the appropriate level of "shading" at each node $\left(0, \frac{1}{2}, \text{ or } 1\right)$, with $v\left[\overline{\text{MIS}}\right] = 3.5$. (A similar representation has been used in Figure (1b) for a feasible solution to the LP relaxation of Problem MIS, with objective function value of 3.) Nemhauser and Trotter (1975) showed that if a variable $x_i = 1$ in any optimal solution to Problem $\overline{\text{MIS}}$, then $x_i = 1$ in *at least* one optimal solution to the above formulation (2.1a)-(2.1d). They also showed that when some such variable x_i is fixed at one, it leads to a significant reduction in the search space for an independent set. Other approaches include the *tree search* algorithm of Loukakis and

Tsouros (1982), and the algorithm proposed by Ebenegger *et al.* (1984) based on the maximization of a pseudo-Boolean function to compute the size of a maximum independent set.

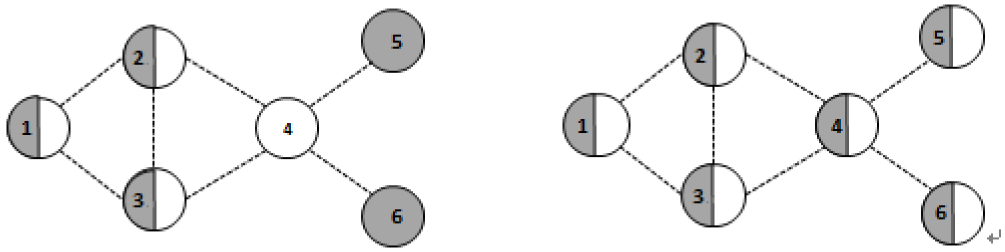


Figure 1(a): $\overline{\text{MIS}}$ optimal solution with obj. value 3.5 **Figure 1(b)**: $\overline{\text{MIS}}$ feasible solution with obj. value 3

Figure 1. Illustrative example displaying an optimal and a feasible solution to Problem $\overline{\text{MIS}}$

Kikusts (1986) proposed a branch-and-bound algorithm for the maximum independent set problem based on a new recursive relation for the independence number of a graph G. Namely,

$$\alpha(G) = \max\left\{1+\alpha\left(G\setminus\left[\{v\}\cup N(v)\right]\right),\ \ \alpha'_v\right\}, \tag{2.2}$$

where $\alpha'_v \equiv \max\left\{|I| :\ I \subseteq G\setminus\{v\} \text{ is an independent set with } \left|I\cap N(v) \geq 2\right|\right\}$. Note that this is different from the recursive relation

$$\alpha(G) = \max\left\{1+\alpha\left(G\setminus\left[\{v\}\cup N(v)\right]\right),\ \ \alpha\left(G\setminus\{v\}\right)\right\}, \tag{2.3}$$

traditionally used in designing branch-and-bound algorithms for the maximum independent set problem. Intuitively, (2.2) provides a stronger relationship than (2.3).

The algorithm of Friden *et al.* (1990) is another branch-and-bound method for the maximum independent set problem, employing tabu search techniques for finding lower bounds. Carraghan and Pardalos (1990) proposed an alternative, simpler approach by examining vertices in *nondecreasing order* of their degree within an implicit (partially) enumerative framework, and showed that this method was extremely efficient, especially for sparse graphs. Bourjolly *et al.* (1997) propose a column generation algorithm, and Rossi and Smriglio (2001) report encouraging computational results with a branch-and-cut algorithm for the maximum independent set problem.

In a pioneering work, Motzkin and Straus (1965) related the independence number to the global optimal value of a certain quadratic function, defined in terms of *continuous variables*, over the standard simplex, as stated below.

THEOREM 1. (Motzkin–Straus). *The global optimal value of the following quadratic program*:

$$\text{Maximize} \quad f(x) = \frac{1}{2} x^\top A_{\bar{G}} x \tag{2.4a}$$

$$\text{subject to:} \quad e^\top x = 1 \tag{2.4b}$$

$$x \geq 0, \tag{2.4c}$$

is given by $\frac{1}{2}\left(1 - \frac{1}{\alpha(G)}\right)$.

The Motzkin-Straus formulation not only yields another approach to determining the maximum independence number, but more importantly, it also provided the first impetus to shift this problem from a *discrete* to a *continuous* setting.

DEFINITION 2. *A clique* C *is a subset of* V *such that the subgraph* $G(C)$ *induced by* C *on* G *is complete.*

The maximum clique problem is to then find a clique of maximum cardinality, and the *clique number* $\omega(G)$ is the cardinality of a maximum clique in G. Recognizing that I is a maximum independent set of G if and only if I is a maximum clique of $\bar{G}$, *i.e.*, $\alpha(G) \equiv \omega(\bar{G})$, a generalization of the Motzkin-Straus theorem was introduced by Bomze (1997) to guarantee a one-to-one correspondence between maximal cliques and local maximizers of a quadratic program, which is summarized in Theorem 2 below.

DEFINITION 3. *For a set of vertices* S, *the characteristic vector of* S, *denoted* x^S, *is a* *vector that satisfies* $x_i^S = 1/|S|$, *if* $i \in S$, *and* $x_i^S = 0$, *otherwise.*

THEOREM 2. (Bomze). *Let* $S \subseteq V$, *and let* x^S *be its characteristic vector. Then the following statements hold:*

- *S is a maximum clique of* G *if and only if* x^S *is a global optimum of the problem:*

$$\text{Maximize}\left\{g(x):\ e^\top x = 1,\ x \geq 0\right\}, \text{where } g(x) = x^\top A_G x + \frac{1}{2}x^\top x. \tag{2.5}$$

In this case, $\omega(G) = \dfrac{1}{2\left(1 - g\left(x^S\right)\right)}$.

- S *is a maximal clique of* G *if and only if* x^S *is a local maximizer of* $g(x)$.
- *All local (and hence global) maximizers of* $g(x)$ *over* S *are strict, and of the form* $x \equiv x^S$ *for some* $S \subseteq V$.

An equivalent continuous formulation to MIS is the *quadratic program* (**QP**) by Shor (1990), where the edge constraint (2.1c) is represented as a *complementarity constraint*, and the binary variables are explicitly represented by their quadratic equalities, to yield the following problem:

$$\text{Maximize} \quad \sum_{i=1}^{n} x_i \tag{2.6a}$$

$$\text{subject to:} \quad x_i x_j = 0, \ \forall\ (i,\ j) \in E \tag{2.6b}$$

$$x_i^2 - x_i = 0, \quad i = 1, \ldots, n. \tag{2.6c}$$

$$0 \leq x_i \leq 1, \quad \forall\, 1 = 1, \ldots, n. \tag{2.6d}$$

He then used this formulation to obtain dual estimates of good quality.

As noted in Balasundaram and Butenko (2005), and subsequently by Pattillo and Butenko (2011), the independence number $\alpha(G)$ also satisfies the following global optimization formulations:

$$\alpha(G) = \max\left\{\sum_{i\in V} x_i \prod_{j\in N(i)} (1-x_j) : 0 \le x_i \le 1, \ \forall i \in V\right\}, \tag{2.7a}$$

$$\alpha(G) = \max\left\{\sum_{i\in V} x_i - \sum_{(i,j)\in E} x_i x_j : 0 \le x_i \le 1, \ \forall\, i \in V\right\}, \quad \text{and} \tag{2.7b}$$

$$\alpha(G) = \max\left\{\frac{\left(\sum_{i\in V} x_i\right)^2}{\sum_{i\in V} x_i^2 + 2\sum_{(i,j)\in E} x_i x_j} : 0 \le x_i \le 1, \ \forall\, i \in V;\ x \ne 0\right\} . \tag{2.7c}$$

Note that, in each of these formulations, the feasible region can be replaced with $x \in \{0,1\}^n$ without changing the optimum objective function value. Formulations (2.7a) and (2.7b) were proved using *probabilistic methods* by Harant *et al*. (1999) and Harant (2000), respectively, and formulation (2.7c) can be obtained using a rather convoluted approach from the Motzkin–Straus theorem. In Abello *et al*. (2001), the authors give *deterministic proofs* that the two continuous formulations (2.7a) and (2.7b) yield the maximum independent set along with generalizations for certain types of dominating sets.

While the authors also show that the Motzkin–Straus theorem (as stated above) can be obtained from these formulations, the syntactically-related polynomial time algorithms presented in the paper only yield *maximal* independent sets. Nevertheless, these algorithms do provide some benchmarking results and are briefly summarized below.

Consider the formulation in (2.7a), where, for any fixed $i \in V$, we can rewrite $F(x)$ in the form:

$$F(x) = (1-x_i)A_i(x) + x_i B_i(x) + C_i(x), \tag{2.8a}$$

where,

$$A_i(x) = \prod_{(i,\ j)\in E} x_j\ , \tag{2.8b}$$

$$B_i(x) = \sum_{(i,\ k)\in E} (1-x_k) \prod_{\substack{(k,\ j)\in E \\ j\ne i}} x_j\ , \tag{2.8c}$$

$$C_i(x) = \sum_{(i,\ k)\notin E} (1-x_k) \prod_{(k,\ j)\in E} x_j\ . \tag{2.8d}$$

Expressions (2.8b)-(2.8d) can be interpreted in terms of the neighborhoods of vertex $i \in V$, where $A_i(x)$ and $B_i(x)$ characterize the first- and second-order neighborhoods of vertex i, respectively, and therefore $C_i(x)$ represents neighborhoods of all vertices, excluding i, which are not characterized by $B_i(x)$. Notice that the fixed variable x_i is absent in (2.8b)-(2.8d), and therefore $F(x)$ is *linear* with respect to each such variable. It is also clear from the above representation that if x^* is an optimal solution of (2.7a), then $x_i^* = 0$ if $A_i(x^*) > B_i(x^*)$, and $x_i^* = 1$ if $A_i(x^*) \le B_i(x^*)$. In this case, $I = \{i \in V : x_i^* = 0\}$ is a maximal independent set, and the following iterative procedure (Algorithm 2) uses this characterization to generate a maximal independent set.

Similarly, we can exploit the properties of formulation (2.7b) to also find maximal independent sets, which is summarized in Algorithm 3 below.

Table 2. Lower bounding heuristic to determine a maximal independent set based on (2.7a)

Algorithm 2. INPUT: $x^0 \in [0,1]^n$; OUTPUT: A maximal independent set $\hat{I}$
Step 1. Set $v := x^0$. Go to Step 2.
Step 2. For $i = 1, \ldots, n$, do: If $A_i(v) > B_i(v)$, then $v_i := 0$,else $v_i := 1$. Go to Step 3.
Step 3. For $i = 1, \ldots, n$, do: If $A_i(v) = 1$, then $v_i = 0$. Go to Step 4.
Step 4. Let $\hat{I} = \{i \in \{1, 2, \ldots, n\} : v_i = 0\}$. Return $\hat{I}$ as a maximal independent set.
END

Table 3. Lower bounding heuristic to determine a maximal independent set based on (2.7b)

Algorithm 3. INPUT: $x^0 \in [0,1]^n$; OUTPUT: A maximal independent set $\hat{I}$

Step 1. Set $v := x^0$. Go to Step 2.

Step 2. For $i = 1, \ldots, n$, do: If $\sum_{(i,j)\in E} v_j < 1$, then $v_i := 1$; else $v_i := 0$. Go to Step 3.

Step 3. For $i = 1, \ldots, n$, do: If $\sum_{(i,j)\in E} v_j = 0$, then $v_i = 1$. Go to Step 4.

Step 4. Let $\hat{I} = \{ i \in \{1, \ldots, n\} : v_i = 1 \}$. Return $\hat{I}$ as a maximal independent set.

END

Several other heuristics for the maximum independent set problem based on both discrete and continuous optimization formulations have been proposed in the literature, (see Avriel *et al*., 1988; Sherali, 1998; and Pardalos *et al*., 1999). Busygin *et al*. (2002) studied the rank-one and rank-two relaxations of the Lovász *semidefinite program* and obtained two continuous formulations for the maximum independent set problem. They used these formulations to develop effective heuristics for finding maximum independent sets and maximum cliques.

Another global optimization formulation that yields the independence number of a graph is the following *fractional programming* (**FP**) formulation:

$$\alpha(G) \quad = \quad \max_{x\in[0,1]^n} \sum_{i\in V} \frac{x_i}{1 + \sum_{j\in N(i)} x_j} \quad . \tag{2.9}$$

Balasundaram and Butenko (2005, 2006) characterize the local maxima of this continuous global optimization formulation and using classical KKT conditions, they deduced several interesting properties of local and global maxima, and false local maxima that exist were eliminated.

Modern integer programming solvers typically combine branch-and-bound strategies with cutting plane methods, efficient preprocessing schemes, including fast heuristics and sophisticated decomposition techniques in order to find an exact solution, and they are often quite effective in determining maximum independent sets of a graph (even with their default settings). However, it is only recently that attempts have been made to solve the independence number problem via *global optimization* approaches. In this paper, we primarily focus on this fractional programming formulation (2.9), and determine the independence number of a graph (and associated independent sets) by designing a specialized global optimization algorithm for this formulation.

There are several important contributions of this research to the literature. First and foremost, as seen in our discussion above, while there are several algorithms that have been designed for determining the independence number of a graph, they are all geared towards exploiting the 0-1 linear programming formulation, namely Problem MIS. On the other hand, there has been considerable work in designing heuristics for continuous formulations that only yield maximal independent sets. In our first contribution, we define a new class of *vertex sets*, and utilize the structure of these vertex sets to propose a simple, yet powerful, construction procedure to determine maximal independent sets. Moreover, we also derive an *explicit characterization* to compute the *number of alternate optima* present in both Problem MIS as well as Problem FP based on these newly defined vertex sets. Several examples that serve to illustrate various concepts and insights are also presented. In our second contribution, while a continuous characterization of the stability number can be found using the Motzkin-Strauss result (Theorem 1), to the best of our knowledge, this paper presents one of the first (exact) globally optimal algorithms for determining the independence number that exploits an underlying continuous formulation. In the course of designing this algorithm, we also present an LP-rounding based heuristic to determine tight lower bounds on the independence number. Furthermore, we relate several of the well-known independence number results (including heuristics, characteristic vectors, etc.) to one another through simple proofs and logical arguments. Third, as seen in our numerical experiments, while the computational time required for solving Problem (2.1a)-(2.1d) is significantly lesser as compared to the computational time required for solving the fractional programming formulation (2.9), we show that the *computational time required per optimal solution* is comparable and in some instances lower for Problem FP as compared to Problem MIS (as the number of alternate optima for Problem FP is significantly greater as compared to Problem MIS). Finally, we also pose two (unproven) conjectures; the first that relates the optimal solution to Problem $\overline{\text{MIS}}$ to the number of alternate optima present in the graph and the second that relates the cardinality of vertex sets to maximal cliques in the graph.

2.2 Properties of the Fractional Programming Formulation

For notational convenience, let $f(x) = \sum_{i \in V} \frac{x_i}{1 + \sum_{j \in N(i)} x_j}$. Then, the fractional programming formulation (2.9) can be expressed as:

$$\textbf{FP: } \alpha(G) = \max_{x \in [0,1]^n} f(x) \equiv \max \left\{ \sum_{i \in V(G)} \frac{x_i}{1 + \sum_{j \in N(i)} x_j} : 0 \le x_i \le 1, \ \forall i \in V \right\}.$$

Then, the following results are evident.

PROPOSITION 1 (Balasundaram and Butenko, 2005). *$f(x)$ is a convex function with respect to x_i, $\forall i \in V$.*

Proof. Consider $f(x)$ in the space of any variable x_k, *i.e.*, x_i is fixed for all $i \in V, i \ne k$.

Then, $f(x)$ can be written as:

$$f(x) = \frac{x_k}{1 + \sum_{j \in N(k)} x_j} + \sum_{i \in N(k)} \frac{x_i}{x_k + 1 + \sum_{j \in N(i) \setminus \{k\}} x_j} + \sum_{i \in V'} \frac{x_i}{1 + \sum_{j \in N(i)} x_j} \quad , \qquad (2.10)$$

where $V' = V \setminus \{k \cup N(k)\}$. Then, the first and second derivatives of $f(x)$ with respect to x_k are:

$$\frac{\partial f}{\partial x_k} = \frac{1}{1 + \sum_{j \in N(k)} x_j} - \sum_{i \in N(k)} \frac{x_i}{\left(x_k + 1 + \sum_{j \in N(i) \setminus \{k\}} x_j \right)^2} \quad , \text{ and} \qquad (2.11a)$$

$$\frac{\partial^2 f}{\partial x_k^2} = \sum_{i \in N(k)} \frac{2x_i}{\left(x_k + 1 + \sum_{j \in N(i) \setminus \{k\}} x_j \right)^3} \geq 0, \quad \forall\ 0 \leq x_i \leq 1, \tag{2.11b}$$

which proves the assertion. ■

THEOREM 3 (Balasundaram-Butenko, 2005). *Every local maxima to FP has a 0-1 solution.*

Proof. From (2.11b), it can be seen that $f(x)$ is a *strictly convex* function with respect to x_k, $\forall\, k \in V$, unless $x_i = 0$, $\forall\, i \in N(k)$. Therefore, $x_k \in \{0, 1\}$ at any local maxima by virtue of a strictly convex function defined over a closed interval attaining its maxima at its bounds. Now, if $x_i = 0$, $\forall\, i \in N(k)$, then, from (2.10), $f(x)$ is a *linear* function with respect to x_k, and for $x_k \in [0,1]$, we can set $x_k = 1$ to attain a local maxima. ■

COROLLARY 1. *If* $\bar{x}_k$ *is fractional at any feasible solution to FP, i.e.,* $0 < \bar{x}_k < 1$, *then either rounding up or down using the following rule always yields an equal or better objective function value.*

If $\sum_{i \in N(k)} \frac{\bar{x}_i}{\left[1 + \sum_{j \in N(i) \setminus \{k\}} \bar{x}_j\right]\left[1 + \bar{x}_k + \sum_{j \in N(i) \setminus \{k\}} \bar{x}_j\right]} < \frac{1}{1 + \sum_{j \in N(k)} \bar{x}_j}$, *then set:* $x_k = \lceil \bar{x}_k \rceil = 1$.

Else, set: $x_k = \lfloor \bar{x}_k \rfloor = 0$.

In so far, while we have studied certain properties of the fractional program (2.9), we now turn our attention towards *characterizing optimal solutions* of both Problem MIS and Problem FP. To begin with, note that Balasundaram and Butenko (2005) presented a deterministic proof that the optimum value of the fractional programming formulation (2.9) indeed yields the independence number. We revisit their proof here as it facilitates our subsequent discussion below.

THEOREM 4 (Balasundaram-Butenko, 2005). *The optimal objective function value of FP* *yields the stability number, i.e.,*

$$\alpha(G) \equiv \max\left\{ f(x) : 0 \le x_i \le 1,\ \forall i \in V \right\}, \textit{ where } \quad f(x) = \sum_{i \in V} \frac{x_i}{1 + \sum_{j \in N(i)} x_j}.$$

Proof. Let f^* denote the optimal objective function value of FP.

<u>Step 1:</u> $\alpha(G) \le f^*$

Let I^* be a maximum independent set, with corresponding 0-1 solution: $x^* \equiv \left(x_i^*,\ \forall\, i = 1,\ldots,V\right)$.

Then, by definition, $\alpha(G) = \left|I^*\right| = \sum_{i \in V} x_i^*$.

Now, the solution $x^* \equiv \left(x_i^*,\ \forall\, i = 1,\ldots,V\right)$ is obviously feasible to FP, and therefore,

$$f^* \ \ge\ f\left(x^*\right) = \sum_{i \in V} \frac{x_i^*}{1 + \sum_{j \in N(i)} x_j^*} = \sum_{i \in V} x_i^* = \alpha(G).$$

<u>Step 2:</u> $\alpha(G) \ge f^*$

Conversely, let $x^* \equiv \left(x_i^*,\ \forall\, i = 1,\ldots,V\right)$ be an optimal 0-1 solution to FP (see Theorem 3).

WLOG, assume that $x_1^* = x_2^* = \cdots = x_r^* = 1$, and $x_{r+1}^* = \cdots = x_n^* = 0$, for some r, $1 \le r \le n$. Now, let $V^* = \{1,\ldots,r\} \subseteq V$, and let $G^* = G\left(V^*,\ E^*\right)$ denote the subgraph induced by V^*. Clearly,

$$f^* = \sum_{i \in V} \frac{x_i^*}{1 + \sum_{j \in N(i)} x_j^*} = \sum_{i \in V^*} \frac{1}{1 + \sum_{j \in N^*(i)} x_j^*} = \sum_{i \in V^*} \frac{1}{1 + \left|N^*(i)\right|}.$$

From the well-known Caro-Wei lower bound (see Section 2.3), we get that

$$\alpha\left(G^*\right) \ge \sum_{i \in V^*} \frac{1}{1 + \left|N^*(i)\right|} \quad \left(= f^*\right),$$

and as $\alpha\left(G\right) \ge \alpha\left(G^*\right) \ge f^*$, and this completes the proof. ■

As noted in the proof of Theorem 4, formulation (2.9) is an unconstrained optimization problem (with only simple variable bounds), and so, any optimal solution corresponding to a maximum independent set I^* is also optimal to the fractional program.

Using this property, the number of *alternate optimal solutions* to both the 0-1 linear programming formulation (2.1a)-(2.1d) and the fractional programming problem (2.9) can be *explicitly characterized*. While there have been several recent attempts made at determining the total number of independent sets in a graph (see Okamoto *et al*. (2008), Zhao (2010), Galvin and Zhao (2011), and Samotij (2015)), our approach is primarily geared towards determining a lower bound on only the total number of maximum independent sets, and in some cases, this bound is tight leading to the exact number of alternate optimal solutions to Problem MIS.

Towards this end, consider the following construction procedure of the duly named *vertex sets* $C^{(k)}$, for all $k \in I$, where I is any independent set (not necessarily maximal/maximum) of G. In this procedure (formalized in Table 4 below), given a specific vertex $k \in I$, delete all nodes $i \in I$, $i \neq k$ and all their neighbors, *i.e*., the corresponding vertex set is given by $C^{(k)} = V \setminus \bigcup_{\substack{i \in I \\ i \neq k}} \left(i \cup N(i) \right)$. Each vertex set $C^{(k)}$ is said to be *generated* by a node $k \in I$, and conversely, node k is said to be the *generating node* for vertex set $C^{(k)}$.

Table 4. Construction of vertex sets *C*^(*k*)

Algorithm 4. INPUT: $G = (V, E)$ and I; OUTPUT: $C^{(k)}$, $\forall\, k \in I$
For $k = 1, \ldots, \alpha(G)$, define: $C^{(k)} = V \setminus \bigcup_{\substack{i \in I \\ i \neq k}} \left(i \cup N(i) \right);$
END

PROPOSITION 2. *Let* I^* *be a maximum independent set of* G. *Given the vertex sets* $C^{(k)}$, $\forall\, k \in I^*$, *as determined by Algorithm 4, the following assertions hold:*

(a) $\left|C^{(k)}\right| \geq 1,\ \forall\, k \in I^{*}$.

(b) $\bigcap_{k \in I^{*}} C^{(k)} \equiv \varnothing$.

(c) $C^{(s)} \cap C^{(t)} \equiv \varnothing,\ \forall\, s, t \in I^{*},\ s \neq t$.

(d) $C^{(k)},\ \forall\, k \in I^{*}$, *is either a singleton node or a clique*.

Proof.

(a) The result is trivially true as each vertex set $C^{(k)}$ contains at least its own generating node k. (Indeed, note that for any generating node $s \in I^{*}$, $s \in C^{(s)}$ and $s \notin C^{(t)},\ \forall\, t \in I^{*},\ s \neq t$.)

(b) We will prove this by contradiction. Suppose that I^{*} is a maximum independent set with $\bigcap_{k \in I^{*}} C^{(k)} = \{p\}$. Then, by virtue of construction of the vertex sets, the edge (k, p) does not exist for all $k \in I^{*}$, and therefore, $I^{*} \cup \{p\}$ must also be an independent set, which contradicts the fact that I^{*} is a maximum independent set.

(c) Suppose that $C^{(s)} \cap C^{(t)} \equiv \{p\}$, for some $s, t \in I^{*},\ s \neq t$. Now, if $(s, p) \in E$, then by construction of vertex sets, $p \notin C^{(t)}$, which is a contradiction. On the other hand, if $p \in C^{(s)}$ and $(s, p) \notin E$, then it must be that the edge (k, p) does not exist for all $k \in I^{*}$, and the result follows from part (b).

(d) We again use a contradiction argument. Consider any $s \in I^{*}$ such that $\left|C^{(s)}\right| \geq 2$, *i.e.*, $C^{(s)}$ is not a singleton set. Now, if $C^{(s)}$ is not a clique, then there exist nodes $p, q \in C^{(s)}$ such that edge $(p, q) \notin G\left(C^{(s)}\right)$, the graph induced by $C^{(s)}$. Moreover, from part (b), we know that $\bigcap_{k \in I^{*}} C^{(k)} = \varnothing$, which implies that $p, q \notin C^{(k)}, \forall\, k \neq s$, and therefore, $I^{*} \cup \{p, q\} \setminus \{s\}$ must also be an independent set of cardinality $\left|I^*\right|+1$, which contradicts the fact that I^* is a maximum independent set. Hence, $C^{(s)}$ must be a clique if $\left|C^{(s)}\right| \geq 2$, or just a singleton node $\{s\}$. ■

Proposition 2 prompts another heuristic approach, where, beginning with any independent set I^0, with $\left|I^0\right| \geq 2$, possibly obtained by using a greedy heuristic (see Algorithm 6 in the following section), a simple construction procedure based on the above defined vertex sets can be iteratively applied to determine maximal independent sets in a graph, which is described in Algorithm 5 below. Figure 2 illustrates this construction procedure to determine a maximal independent set, where beginning with the independent set $I = \{4, 5\}$, the algorithm progressively computes the vertex sets, and updates the nodes to be included in the maximal independent set in an iterative fashion, until the termination criterion is satisfied.

Table 5. A construction procedure for determining a maximal independent set.

Algorithm 5. INPUT: $G = (V, E)$ and I^0; OUTPUT: A maximal independent set $\hat{I}$

Step 1. Set $\hat{I} := I^0$. Go to Step 2.

Step 2. Using Algorithm 4, construct the vertex sets $C^{(k)}, \forall\, k \in \hat{I}$. Go to Step 3.

Step 3. If $\bigcap_{k \in \hat{I}} C^{(k)} = \varnothing$, stop; and return $\hat{I}$ as a maximal independent set. Else, go to Step 4.

Step 4. Select any node $p \in \bigcap_{k \in \hat{I}} C^{(k)}$, update $\hat{I} \leftarrow \hat{I} \cup \{p\}$. Go to Step 2.

END

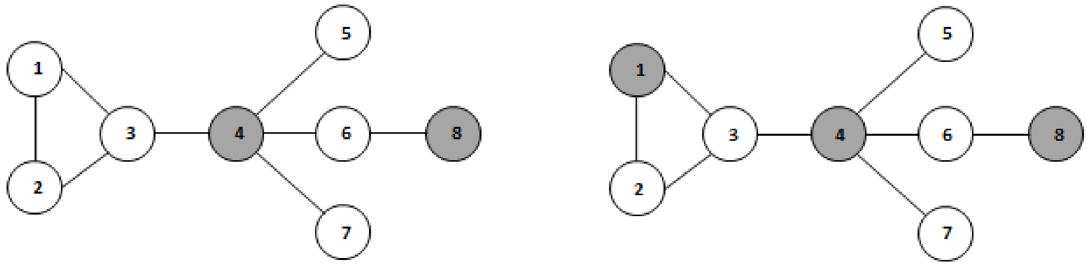


Figure 2(a): $C^{(4)}$ = {1, 2, 3, 4, 5, 7} and $C^{(8)}$ = {1, 2, 8}   **Figure 2(b)**: $C^{(1)}$ = {1, 2}; $C^{(4)}$ = {4, 5, 7}; $C^{(8)}$ = {8}; RETURN {1, 4, 8}

Figure 2. Illustrative example displaying the construction procedure of a maximal independent set

Now, in order to characterize the alternate optimal solutions to Problem MIS as well as Problem FP, we need to first refine the vertex sets obtained from Algorithm 4. Consider a maximum independent set I^*, and let $C^{(k)}, \ \forall k \in I^*$ be the derived vertex sets. Define $C \equiv \bigcup_{k \in I^*} C^{(k)}$, and construct an induced (virtual) graph $G(C) \equiv (\tilde{V}^0, \tilde{E}^0)$, where each set $C^{(k)}, \ \forall k \in I^*$, represents a "dummy" node of $\tilde{V}^0$, and $\tilde{E}^0$ comprises only "inter-set" edges, *i.e.*, $\tilde{E}^0 = \{(p,q) \in E \mid p \in C^{(s)}, q \in C^{(t)}, \forall s,t \in I^*, s \neq t\}$. If $\tilde{E}^0 \equiv \varnothing$, then we are done; otherwise, delete nodes from vertex sets in nonincreasing order of degree until the underlying dummy graph resulting from the updated vertex sets is rendered edgeless. This refinement procedure is formally outlined in Algorithm 6, and illustrated with an example in Figure 3 below.

Table 6. A refinement procedure for construction of disjoint vertex sets $\tilde{C}^{(k)}$.

Algorithm 6. INPUT: $G(C) \equiv (\tilde{V}^0, \tilde{E}^0)$; OUTPUT: $\tilde{C}^{(k)}, \ \forall k \in I^*$

Step 1. Set $\tilde{C}^{(k)} := C^{(k)}, \ \forall k \in I^*$. Go to Step 2.

Step 2. Construct $G(\tilde{C}) = (\tilde{V}, \tilde{E})$, where $\tilde{C} \equiv \bigcup_{k \in I^*} \tilde{C}^{(k)}$. If $\tilde{E} \equiv \varnothing$, stop; and return $\tilde{C}^{(k)}, \ \forall k \in I^*$. Else, go to Step 3.

Step 3. Select a node with maximum degree in $G(\tilde{C})$, say $p \in \tilde{C}^{(k)}$. Update $\tilde{C}^{(k)} \leftarrow \tilde{C}^{(k)} \setminus \{p\}$, and delete the edges corresponding to node p from $\tilde{E}$. Go to Step 2.

END

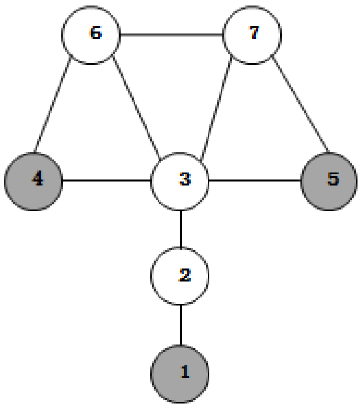


Figure 3(a): Graph with I^* = {1, 4, 5}

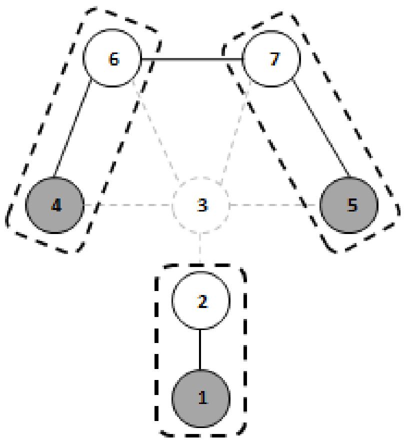


Figure 3(b): Virtual graph formed with vertex sets: $C^{(1)}$ = {1, 2}, $C^{(4)}$ = {4, 6}, $C^{(5)}$ = {5, 7}

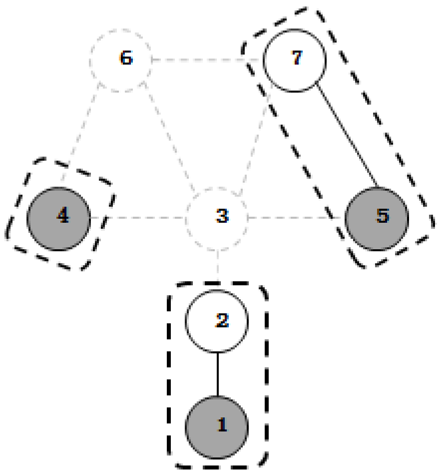


Figure 3(c): Refined vertex sets and edgeless dummy graph formed after deleting node {6} and edge (6, 7)

Figure 3. Illustrative example displaying the refinement procedure of vertex sets to induce an edgeless graph

REMARK 2. As the refinement procedure outlined above begins by deleting nodes in the nonincreasing order of degree, singleton vertex sets $C^{(k)}$ are not affected, and therefore no singleton vertex set $C^{(k)}$ gets eliminated by the refinement procedure, *i.e.*, $\tilde{C}^{(k)} \neq \varnothing$, $\forall\, k \in I^*$. Furthermore, the resulting vertex sets retain the property that they are either singleton nodes or cliques.

THEOREM 5. *Given a maximum independent set* I^* *and the corresponding (refined) sets* $\tilde{C}^{(k)}$, $\forall\, k \in I^*$. *Then,*

(a) *The number of alternate optimal solutions to Problem MIS, denoted by* Λ_{MIS}, *is greater than or equal to* $\prod_{k\in I^*}\left|\tilde{C}^{(k)}\right|$.

(b) *The number of alternate optimal solutions to Problem FP, denoted by* Λ_{FP}, *is greater than or equal to* $\prod_{k\in I^*}\left(2^{\left|\tilde{C}^{(k)}\right|} - 1\right)$.

Proof.

(a) As $\bigcap_{k\in I^*} \tilde{C}^{(k)} \equiv \varnothing$, and $G\left(\tilde{C}\right)$ is edgeless, where $\tilde{C} \equiv \bigcup_{k\in I^*} \tilde{C}^{(k)}$, a maximum independent set can be formed by (arbitrarily) selecting one node from each set $\tilde{C}^{(k)}$, $\forall\, k \in I^*$. Thus, the number of maximum independent sets in the graph is bounded below as $\Lambda_{\text{MIS}} \geq \prod_{k\in I^*}\left|\tilde{C}^{(k)}\right|$.

(b) As the fractional programming formulation (2.9) only has simple variable bounding constraints, and $\bigcap_{k\in I^*} \tilde{C}^{(k)} \equiv \varnothing$, any arbitrarily chosen (clique) subset of $\tilde{C}^{(k)}$, $\forall\, k \in I^*$, is optimal, with each $\tilde{C}^{(k)}$ contributing a value of *one* to the optimum objective function value. There are $\left(2^{\left|\tilde{C}^{(k)}\right|} - 1\right)$ possible subsets (with the only exclusion being the null set) that can be chosen from each $\tilde{C}^{(k)}$, $\forall\, k \in I^*$, and hence, the total number of alternate optimal solutions to the fractional programming formulation (2.9) is $\Lambda_{\text{FP}} \geq \prod_{k\in I^*}\left(2^{\left|\tilde{C}^{(k)}\right|} - 1\right)$. ■

COROLLARY 2. *The optimal solution to FP corresponds to a maximum independent union* *of cliques, with each clique contributing a value of one to the optimum objective function value.* ■

To illustrate Theorem 5 and Corollary 2, consider the same graph shown in Figure 1, where one possible choice of a maximum independent set is $I^* = \{1, 5, 6\}$, with $\alpha(G) = 3$. In this case, the corresponding vertex sets are given by: $C^{(1)} = \{1, 2, 3\}$, $C^{(5)} = \{5\}$, and $C^{(6)} = \{6\}$. It is easy to verify that the vertex sets satisfy conditions (a), (b), and (c) from Proposition 2; Corollary 2 is valid; and moreover, the virtual graph induced by these vertex sets is edgeless. Therefore, from Theorem 5, a lower bound on the number of alternate optima to Problem MIS (or equivalently, the number of maximum independent sets in this graph) and Problem FP can be respectively determined as $\prod_{k \in I^*} \left| C^{(k)} \right| = 3 \times 1 \times 1 = 3$ and $\prod_{k \in I^*} \left(2^{\left| C^{(k)} \right|} - 1 \right) = \left(2^3 - 1\right) \times \left(2^1 - 1\right) \times \left(2^1 - 1\right) = 7$. Note that all solutions to Problem MIS are feasible to Problem FP, but not vice versa; so, there are four additional optimal solutions to Problem FP as compared to Problem MIS in this case. Figures (4a)-(4c) illustrate all possible alternate optima to Problem MIS, and Figures (4d)-(4g) round out the remaining (additional) solutions to Problem FP. Furthermore, as seen in Figure 4, optimal solutions to Problem FP are not necessarily maximum independent sets, but they are a *mutually independent union* of either singleton nodes or cliques, with each clique contributing a value of one to the optimum objective function value, and every member of a clique contributing an equal amount. For example, Figure (4g) displays the clique $\{1, 2, 3\}$ being part of the optimal solution, with each node in this clique contributing a value of $\frac{1}{3}$ (as indicated by its shaded part in this clique).

As another example, for the graph given in Figure 3, only a strict lower bound on the number of independent sets can be obtained by using Theorem 5, which is given by $\Lambda_{\text{MIS}} \geq 4$. (Note that the total number of alternate optima to Problem MIS in this case is actually equal to six.) A similar computation can be done for Problem FP in this case.

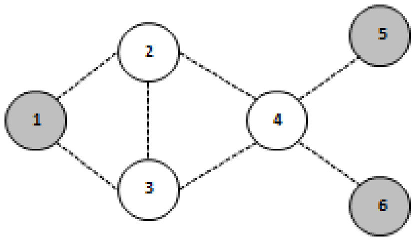


Figure 4(a): Optimal solution with I^* = {1, 5, 6}

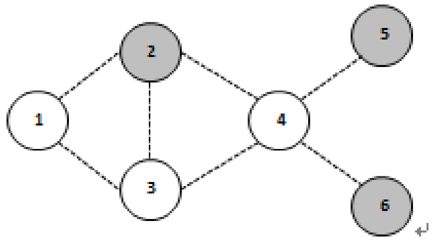


Figure 4(b): Optimal solution with I^* = {2, 5, 6}

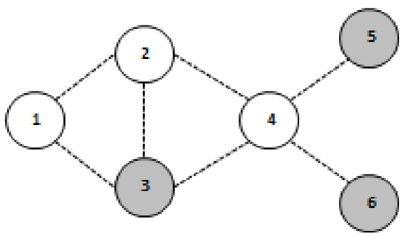


Figure 4(c): Optimal solution with I^* = {3, 5, 6}

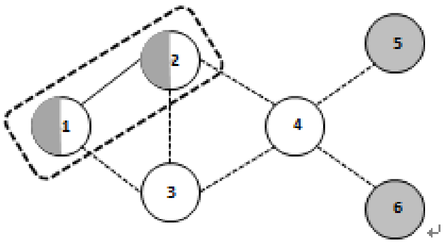


Figure 4(d): Optimal solution with FP^1 = {1, 2, 5, 6}

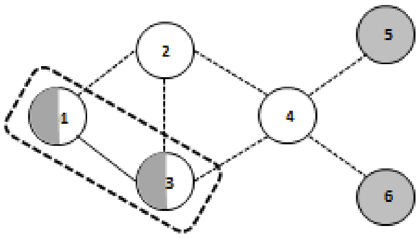


Figure 4(e): Optimal solution with FP^1 = {1, 3, 5, 6}

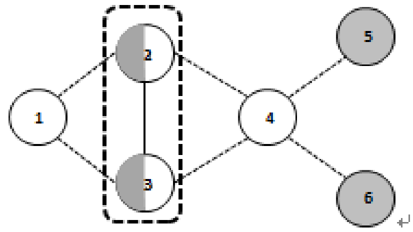


Figure 4(f): Optimal solution with FP^1 = {2, 3, 5, 6}

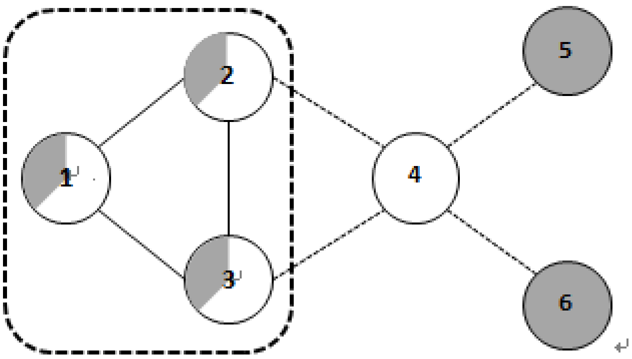


Figure 4(g): Optimal solution with FP^1 = {1, 2, 3, 5, 6}

Figure 4. Illustrative example displaying all possible alternate optimal solutions to Problem MIS and Problem FP

An Alternate Approach to Obtaining the Lower Bound

Theorem 5 provides a characterization to compute lower bounds on the number of alternate optima present in both Problem MIS and Problem FP. During the refinement procedure employed to generate vertex sets, if the number of deleted vertices (see Step 3 in Algorithm 6) is large, then the lower bounds obtained can be rather weak. However, the lower bounds can also be obtained by following an *edge deletion* procedure on the artificial graph $G\left(\tilde{C}\right)=\left(\tilde{V},\tilde{E}\right)$ rather than the node deletions advocated in Theorem 5.

Specifically, suppose that the edge $\left(i,\, j\right)\in\tilde{E}$, where $i\in C^{(s)}$ and $j\in C^{(t)}$. Then, an improved lower bound on Problem MIS can be obtained by using:

$$\Lambda_{\text{MIS}} \ \geq\ \max\left\{0,\ \prod_{k\in I^*}\left|C^{(k)}\right| \ -\ \sum_{\substack{(i,\,j)\in\tilde{E}\\ i\in C^{(s)},\ j\in C^{(t)}}}\left(\prod_{k\in I^*:k\neq s,t}\left|C^{(k)}\right|\right)\right\}\ , \qquad (2.12a)$$

where, if $I^*=\left\{s,\,t\right\}$, *i.e.*, $\left\{k\in I^*:k\neq s,t\right\}\equiv\varnothing$, we define $\prod_{k\in I^*:k\neq s,t}\left|C^{(k)}\right|\equiv 1$. Moreover, note that if there are multiple edges in the virtual graph connecting the same vertex sets, then the second term in Equation (2.12a) would delete combinations by randomly selecting any one such edge, and not perform repeated (identical) deletions. In other words, each *vertex set pair* $\left(C^{(s)},C^{(t)}\right)$ is selected exactly once in Equation (2.12a).

Similarly, an improved lower bound on the number of alternate optimal solutions to Problem FP can be represented as:

$$\Lambda_{\text{FP}} \ \geq\ \max\left\{0,\ \prod_{k\in I^*}\left(2^{\left|C^{(k)}\right|}-1\right) - \sum_{\substack{(i,\,j)\in\tilde{E}\\ i\in C^{(s)},\ j\in C^{(t)}}}\left\{\left(\prod_{k\in I^*:k\neq s,t}\left(2^{\left|C^{(k)}\right|}-1\right)\right)\left(2^{\left|C^{(s)}\right|-1}\right)\left(2^{\left|C^{(t)}\right|-1}\right)\right\}\right\} \qquad (2.12b)$$

Indeed, while (2.12) removes all possible combinations by deleting a *single* edge, it is evident that we are *over-estimating* the number of infeasible combinations, and therefore, to obtain the exact number of independent sets in the graph we would need to employ (some case-specific variants of) the *set-theoretic principle of inclusion-exclusion* where we: (i) delete the combinations corresponding to a single edge; (ii) add combinations corresponding to pairwise selections of edges, selected from mutually disjoint vertex sets; (iii) delete combinations with 3-tuples of edges; and so on, until all possibilities are exhausted, leading to an exact characterization and count of the number of alternate optima present in the graph. However, for most practical purposes, the bounds given in either Theorem 5 or Equation (2.12a) are sufficient to provide a good estimate on the number of maximum independent sets in the graph, and therefore, such higher order additions/deletions are not further explored in our work.

Numerically, applying the inequalities (2.12a) and (2.12b) on the graph given in Figure 3, we get that $\Lambda_{\text{MIS}} \equiv 8 - 2 = 6$ and $\Lambda_{\text{FP}} \equiv 27 - 3\times 2\times 2 = 15$, which are exactly the number of alternate optima to Problem MIS and Problem FP, respectively.

Exceptions to Obtaining the Exact Number of Alternate Optima

The above defined procedure to compute the exact number of alternate optimal solutions to Problem MIS fails for certain categories of graphs, notably when applied to those graphs that exhibit a high level of *symmetry*. In such cases, the product $\prod_{k\in I^*} \left|C^{(k)}\right|$ *underestimates* the number of independent sets, and therefore, neither the refinement procedure (Algorithm 6) nor the improvement in lower bound (2.12a), both of which only work towards further reducing the bound, yield better results.

For example, consider the graph given in Figure 5(c), which has several parallel paths, and in this case, the vertex sets generated by any of the maximum independent sets do not result in the exact number of alternate optimal solutions. Conversely, for the graph given below in Figure 5(a), where beginning with the maximum independent set $\{1, 4, 7\}$, and applying equation (2.12a), we get $\Lambda_{\text{MIS}} \geq 8$, which is tight; however, if the originally chosen independent set is $\{2, 4, 7\}$, seen in Figure 5(b), the method yields a weaker lower bound given by $\Lambda_{\text{MIS}} \geq 6$. Hence, it is evident that the initial choice of the generating maximum independent set is a very important component in obtaining a good estimate on the number of alternate optimal solutions to Problem MIS.

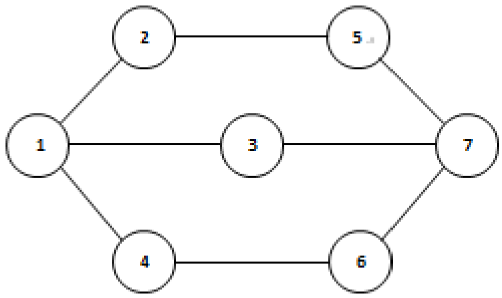


Figure 5(c): Graph for which no maximum independent set yields a tight bound on Λ_{MIS}.

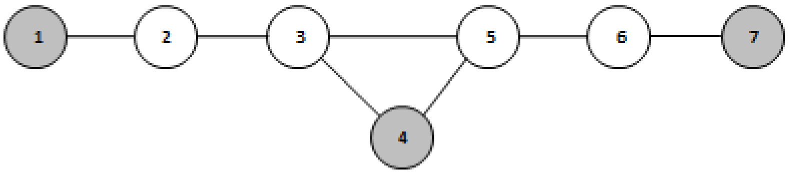


Figure 5(a): Maximum independent set I^* = {1, 4, 7}, with Λ_{MIS} = 8.

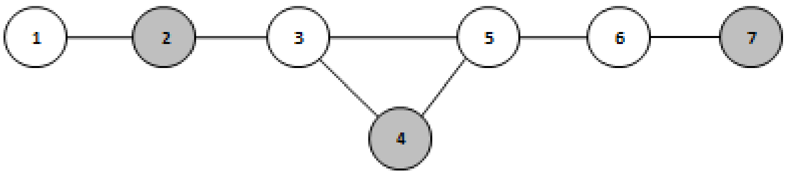


Figure 5(b): Maximum independent set I^* = {2, 4, 7}, with Λ_{MIS} = 6.

Figure 5. Examples of highly symmetric graphs for which an exact bound on the number of alternate optimal solutions might not be obtained.

Nevertheless, even in such instances, Theorem 5 and Equation (2.12) provide valid lower bounds, and serve as reasonable estimates on the number of alternate optima for both Problem MIS and Problem FP. Summarizing all of the ideas presented in the above discussion, Table 7 lists the lower bounds computed using Theorem 5, Equation (2.12), and the exact number of alternate optima for the graphs shown in Figures 3(a), 5(a), and 5(b). As seen from the results, it is evident that, for these examples, the bounds obtained from the edge deletion procedure, namely Equation (2.12), are better than the ones obtained from the node deletions advocated in Theorem 5, and in some instances, the bounds are tight. However, note that this trend does not hold true in general, as seen from our computations on larger graphs.

Table 7: Estimates on the number of alternate optima using various methods

	Figure 3(a)		Figure 5(a)		Figure 5(b)	
	MIS	FP	MIS	FP	MIS	FP
Theorem 5	4	9	4	9	4	9
Equation 12	6	15	8	15	6	15
Exact Number of Alternate Optima	6	15	8	23	6	15

In closing this Section, as noted in Theorem 5, each vertex set $\tilde{C}^{(k)}$, $\forall\, k \in I^*$, contributes a value of one to the objective function of Problem FP, and moreover, given any (clique) subset $S^{(k)} \subseteq \tilde{C}^{(k)}$, every node in $S^{(k)}$ contributes an *equal* amount, *i.e.*, $\frac{x_i}{1+\sum_{\substack{j\in N(i)\\ j\in S^{(k)}}} x_j} = \frac{1}{\left|S^{(k)}\right|}$, $\forall\, i \in S^{(k)}$, and furthermore, $\sum_{i\in S^{(k)}} \frac{x_i}{1+\sum_{\substack{j\in N(i)\\ j\in S^{(k)}}} x_j} = 1$. Interestingly enough, this solution can be correlated to the characteristic vector discussed in Theorem 2. Denoting $\tilde{C} \equiv \bigcup_{k=1}^{\alpha(G)} \tilde{C}^{(k)}$, it can be easily verified that the characteristic vector $x^{\tilde{C}}$, whose components are: $x_i^{\tilde{C}} = \frac{1}{\sum_{k=1}^{\alpha(G)} \left|\tilde{C}^{(k)}\right|}$, if $i \in \tilde{C}^{(k)}$, for some $k \in I^*$, and 0 otherwise; yields $\omega\left(\bar{G}\right) = \frac{1}{2\left(1-g\left(x^{\tilde{C}}\right)\right)}$, where $g(x)$ is as defined in (2.5).

Motivated by these insights, we are now ready to present a reformulation of Problem FP, and subsequently prescribe a specialized global optimization algorithm for solving the reformulated problem, while exploiting the underlying properties (as seen in this section) that are inherent to this formulation.

2.3 A Global Optimization Algorithm for Solving Problem FP

To make Problem FP more amendable to algorithmic manipulations, let,

$$y_i = 1 + \sum_{j \in N(i)} x_j , \quad \forall\, i \in V , \tag{2.13}$$

which yields the following equivalent representation:

FP-1: Maximize $\sum_{i \in V} \frac{x_i}{y_i}$ (2.14a)

subject to: $y_i = 1 + \sum_{j \in N(i)} x_j , \quad \forall\, i \in V$ (2.14b)

$0 \le x_i \le 1, \ 1 \le y_i \le 1 + d_i \ , \quad \forall\, i \in V ,$ (2.14c)

where, for the sake of notational convenience, we have represented the cardinality of the set $N(i)$ as d_i, *i.e.*, $|N(i)| \equiv d_i$.

Recognizing that Problem FP-1 is linear with the exception of the objective function, we linearize the objective function by defining an auxiliary variable,

$$z_i = \frac{x_i}{y_i} \ , \quad \forall\, i \in V , \tag{2.15}$$

which yields the following restatement of the problem:

FP-2: Maximize $\sum_{i \in V} z_i$ (2.16a)

subject to: $y_i = 1 + \sum_{j \in N(i)} x_j , \quad \forall\, i \in V$ (2.16b)

$x_i = y_i z_i \ , \quad \forall\, i \in V$ (2.16c)

$0 \le x_i \le 1, \ 1 \le y_i \le 1 + d_i , \ 0 \le z_i \le 1, \quad \forall\, i \in V ,$ (2.16d)

where we have explicitly introduced the *bilinear* constraint (2.16c), which is the cross-multiplied variant of constraint (2.15). Note that the only complicating term in Problem (2.16a)-(2.16d) is the nonlinear constraint (2.16c). In order to generate an equivalent linear formulation, we construct the convex and concave outer-envelopes of the bilinear function $y_i z_i$ over its corresponding bounding rectangle to obtain the equivalent *Reformulated Fractional Program* (**RFP**):

$$\textbf{RFP:} \quad \text{Maximize} \quad \sum_{i \in V} z_i \tag{2.17a}$$

$$\text{subject to:} \quad y_i = 1 + \sum_{j \in N(i)} x_j, \quad \forall\, i \in V \tag{2.17b}$$

$$-x_i + z_i \le 0, \quad \forall\, i \in V \tag{2.17c}$$

$$-x_i + \left(1 + d_i\right) z_i \ge 0, \quad \forall\, i \in V \tag{2.17d}$$

$$-x_i + z_i + y_i \ge 1, \quad \forall\, i \in V \tag{2.17e}$$

$$-x_i + y_i + \left(1 + d_i\right) z_i \le \left(1 + d_i\right), \quad \forall\, i \in V \tag{2.17f}$$

$$x_i \in \{0,1\}, \; 1 \le y_i \le 1 + d_i \text{ and integer}, \; 0 \le z_i \le 1 \quad \forall\, i \in V, \tag{2.17g}$$

where, constraints (2.17c)-(2.17g) impose $x_i = y_i z_i$ at the optimum, and we have explicitly introduced the binary restrictions on the x-variables (based on Theorem 3), which in turn results in integer restrictions on the y-variables.

REMARK 2. In obtaining the reformulated (linearized) Problem RFP, we have utilized the McCormick (1976) inequalities, which form the convex/concave outer envelopes of the bilinear term $x_i = y_i z_i$ over its corresponding bounding rectangle. Alternately, we could have employed convex underestimators directly for the linear fractional term in (2.15), but such an approach would result in a far more complicated convexification process involving certain nonlinear (but convex) constraints, which would then need to be subsequently linearized or handled via a convex programming solver. Further details on deriving such reformulations can be found in Quesada and Grossmann (1995), Zamora and Grossmann (1998), and Tawarmalani and Sahinidis (2000).

2.3.1 Lower and Upper Bounds

Note that RFP can be solved by using a specialized branch-and-bound algorithm (as delineated in the discussion below). However, the effectiveness of such an approach is highly influenced by the presence of valid lower and upper bounds for the problem. Hence, as a precursor to the branch-and-bound approach, we devise heuristic methods that yield *tight* lower and upper bounds on the problem.

Now, the Caro-Wei's inequality (see Abello *et al*., 2001), which gives a valid *lower bound* on the independence number of a graph, is given by:

$$\alpha(G) \geq \sum_{i \in V} \frac{1}{1+|N(i)|} \quad . \tag{2.18}$$

It can be easily seen that the Caro-Wei lower bound is trivially obtained by setting $x_i \equiv 1$, $\forall\, i \in V$, in the objective function of Problem FP. Also, it is well-known that while this lower bound satisfies: $\alpha(G) \geq \sum_{i \in V} \frac{1}{1+|N(i)|} \geq \sum_{i \in V} \frac{1}{(1+\bar{d})} \geq \sum_{i \in V} \frac{1}{(1+\Delta)} = \frac{n}{(1+\Delta)}$, this inequality is very weak in practice, and moreover, it is not necessarily associated with any independent set. Hence, in order to derive *feasible* independent sets at a relatively low computational cost, we utilize the well-known *degree-based greedy heuristic* (see Algorithm 7 below). We prove that this greedy heuristic not only determines a lower bound on the problem that is always greater than or equal to the Caro-Wei's lower bound, but it also provides us with a feasible solution to MIS, and correspondingly Problem FP as well.

LEMMA 2. *The lower bound on the independence number obtained from the degree-based greedy heuristic (Algorithm 7) is always greater than or equal to the Caro-Wei lower bound*.

Proof. At some iteration $q \geq 0$ of the algorithm, suppose that node i is selected for inclusion into the set L_2^q. Clearly, a node $i \in L_2^q$ contributes a value of *one* to the lower bound obtained from Algorithm 5. Hence, it is sufficient to prove that

$$\frac{1}{1+|N(i)|} + \sum_{k \in N(i)} \frac{1}{1+|N(k)|} \leq 1, \tag{2.19}$$

where the LHS in (2.19) is the contribution of node *i* and all of its (remaining) neighbors at stage *q* to the Caro-Wei lower bound. As we select nodes in nondecreasing order of degree, if node *i* is selected at stage *q*, it must be that $|N(i)| \leq |N(k)|, \ \forall\, k \in L_1^q$. Therefore,

$$\frac{1}{1+|N(i)|} + \sum_{k \in N(i)} \frac{1}{1+|N(k)|} \leq \frac{1}{1+|N(i)|} + \sum_{k \in N(i)} \frac{1}{1+|N(i)|} = 1,$$

which completes the proof. ■

Table 8: Lower bounding degree-based greedy heuristic

Algorithm 7. Degree-based Lower Bounding Greedy Heuristic

Step 0. Arrange the nodes in $G = (V, E)$ in nondecreasing order of degree, *i.e.*, $\{0 \leq d_{i_1} \leq d_{i_2} \leq \cdots \leq d_{i_n} \equiv \Delta\}$. Go to Step 1.

Step 1. Initialize $q = 0$, $L_1^q = \{i_1, ..., i_n\}$ and $L_2^q = \varnothing$. Go to Step 2.

Step 2. If $L_1^q = \varnothing$, stop; return $|L_2^q|$ as a lower bound on the independence number. Else, go to Step 3.

Step 3. Select the node with smallest degree (from the left) in L_1^q, say i_k, and go to Step 4.

Step 4. Update $L_2^{q+1} \leftarrow L_2^q \cup \{i_k\}$ and $L_1^{q+1} \leftarrow L_1^q \setminus \{i_k \cup N(i_k)\}$. Go to Step 5.

Step 5. Update $q \leftarrow q+1$, and go to Step 2.

In a similar vein to Algorithm 7, another lower bound on the independence number that can be computed relatively quickly is based on applying a simple *rounding heuristic* to $\bar{x} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$, an optimal solution to Problem $\overline{\text{MIS}}$ (or alternately, Problem $\overline{\text{RFP}}$). Let $V^0 = \{i : \bar{x}_i = 0\}$, $V^1 = \{i : \bar{x}_i = 1\}$, and $V^{\frac{1}{2}} = \{i : \bar{x}_i = \frac{1}{2}\}$, with $V^0 \cup V^{\frac{1}{2}} \cup V^1 \equiv V$ and $V^0 \cap V^{\frac{1}{2}} \cap V^1 \equiv \varnothing$. Then, the following procedure (Algorithm 8) also yields an independent set.

Table 9: LP rounding-based lower bounding heuristic

Algorithm 8. LP Rounding-based Lower Bound Heuristic

Step 0. Arrange the nodes in $V^{\frac{1}{2}} = \{i : \bar{x}_i = \frac{1}{2}\}$ in nondecreasing order of their degree, *i.e.*, $1 \le d_{i_1} \le d_{i_2} \le \cdots \le d_{i_n}$. Go to Step 1.

Step 1. Initialize $q = 0$, $V_q^0 = \{i_1, \ldots, i_n\}$ and $V_q^1 = \varnothing$. Go to Step 2.

Step 2. If $V_q^0 = \varnothing$, stop; return $\left|V^1 \cup V_q^1\right|$ as a lower bound on the independence number. Else, go to Step 3.

Step 3. Select the node with smallest degree (from the left) in V_q^0, say i_k, set $x_{i_k} = 1$, and go to Step 4.

Step 4. Update $V_{q+1}^1 \leftarrow V_q^1 \cup \{i_k\}$ and $V_{q+1}^0 \leftarrow V_q^0 \setminus \{i_k \cup N(i_k)\}$. Go to Step 5.

Step 5. Update $q \leftarrow q+1$, and go to Step 2.

REMARK 3. It can be easily verified that using the feasible solution to Problem $\overline{\text{MIS}}$, outlined in Remark 1, as a starting point in Algorithm 8 (with ties broken in the nondecreasing order of their degree) yields the *same solution* as the degree-based greedy heuristic. Therefore, Algorithm 7 is indeed a special case of Algorithm 8, and furthermore, it can be easily verified that the cardinality of the independent set obtained by Algorithm 8 is always greater than or equal to the cardinality of the independent set obtained by Algorithm 7, albeit at a greater computational effort.

Conversely, the theoretically *best* (or *smallest*) *upper bound* on the independence number that can be determined in polynomial time is the *Lovász theta function* (Lovász, 1979), which can be characterized as the solution to the following *semidefinite program* (**SDP**):

$$\alpha(G) \le \vartheta(G) \equiv \max\left\{e^\top Xe : \ Trace(X) = 1,\ X_{ij} = 0,\ \forall (i,\ j) \in E,\ X \succeq 0\right\} \quad (2.20)$$

Recognizing the difficulty in determining $\vartheta(G)$ as SDPs are computationally intensive to solve, particularly for large-scale instances, an easier-to-compute upper bound that is advocated is the objective function value of the following *convex program* (Luz and Schrijver, 2005):

$$\alpha(G) \leq \vartheta(G) \leq \upsilon(G) \equiv \max\left\{2e^{\top}x - x^{\top}(H+I)x:\ x \geq 0\right\}, \qquad (2.21)$$

where $H = \dfrac{1}{-\lambda_{\min}(A_G)} A_G$ (recall that A_G is the adjacency matrix of graph G), and $\lambda_{\min}(A_G)$ is the minimum eigenvalue of A_G.

Summarizing all of the above discussion, in our computations, we compare the lower bounds obtained from: (i) maximal independent set heuristics (Algorithms 2, 3, and 5); (ii) greedy and rounding heuristics (Algorithms 7 and 8); and (iii) simple rounding rule (Corollary 1). Moreover, at each stage of the search process, we judiciously employ one of these lower bounding mechanisms and pick the best lower bound available to aid the search process and curtail the size of the enumeration tree. Similarly, we use the upper bounds obtained from: (i) convex relaxation (21); and (ii) the objective function value of Problems $\overline{\text{MIS}}$ and $\overline{\text{RFP}}$ (where $\overline{\text{RFP}}$ denotes the linear programming relaxation of Problem RFP) rounded below to its nearest integer; to reduce the size of the search tree.

2.3.2 Branch-and-Bound Algorithm

For the sake of convenience, we shall denote the set of variables in RFP, with obvious vector notation, as $\xi \equiv (x, y, z)$, and let $\alpha^{LP}(G)$ represent the optimum objective function value of Problem $\overline{\text{RFP}}$ (possibly rounded down to its nearest integer). Then, the following result is evident.

LEMMA 1. *$\alpha^{LP}(G)$ provides an upper bound on $\alpha(G)$. Moreover, if $\bar{\xi} = (\bar{x}, \bar{y}, \bar{z})$ solves $\overline{\text{RFP}}$ and if this solution satisfies constraints (2.16c), then it is optimal to (2.16) with the same objective function value.*

Proof. Obvious from construction. ■

Lemma 1 and the above discussion on lower and upper bounds prompt a branch-and-bound algorithm for solving FP via RFP and its LP (continuous) relaxation, $\overline{\text{RFP}}$. In this algorithm, we adopt a branch-and-bound process based on partitioning the search space by imposing integer restrictions on the y-variables in the problem. Given any node sub-problem associated with the branch-and-bound tree, we solve the corresponding LP relaxation. Let $\bar{\xi} = (\bar{x}, \bar{y}, \bar{z})$ be the optimal solution obtained for this node subproblem. If the condition of Lemma 1 holds true, then we will also have solved the node subproblem of FP. Otherwise, we could apply any one of Algorithms 2, 3, or 8, or simply Corollary 1, to (possibly) update the incumbent solution, and branch at this node by partitioning the search space as follows.

Branching Rule. Find the maximum discrepancy: $i^* \in \arg\max_{i \in V} |\bar{x}_i - \bar{y}_i \bar{z}_i|$. Then, for the corresponding variable i^* that achieves this maximum value, impose the dichotomy $x_{i^*} = 0$ and $x_{i^*} = 1$ to generate the children nodes, breaking ties in favor of the variable with the least degree.

At each stage t of this branch-and-bound procedure, $t = 0, 1, \ldots, T$, we will have a set of non-fathomed or *active nodes* A_t, where each node $a \in A_t$ is indexed by a set of *fixed variables*. (To initialize, at $t = 0$, the set $A_0 = \{0\}$, with none of the variables being fixed.) For each node $a \in A_t$, an upper bound UB_a will be given by $\overline{\text{RFP}}^a$, where, in our notation, $\overline{\text{RFP}}^0 \equiv \overline{\text{RFP}}$. As a result, the (*global*) *upper bound* at stage *t* for RFP (and equivalently, FP) is given by

$$\text{UB} \equiv \underset{a \in A_t}{\text{maximum}} \{\text{UB}_a\}. \tag{2.22}$$

Whenever any upper bounding node subproblem is solved, we can apply Algorithm 6 or Corollary 1 in order to derive a lower bound (and possibly update the incumbent solution) on the overall problem FP, where the corresponding lower bound value is given by (2.17a). Accordingly, let $\xi^* = \left(x^*, y^*, z^*\right)$ be the best such incumbent solution found, having an objective value of ν^*. Naturally, whenever $UB_a \le \nu^*$, we can fathom node *a*. (Practically, we can fathom node *a* whenever $UB_a \le \nu^*\left(1+\varepsilon\right)$, for some optimality tolerance $\varepsilon > 0$.) Hence, the active nodes at any stage *t* would satisfy $UB_a > \nu^*,\ \forall\, a \in A_t$. From this set of active nodes, we now select a node $a(t)$ that yields the greatest upper bound, *i.e.*, for which $UB_{a(t)} \equiv \text{UB}$ as given by (2.22). Note that for the corresponding solution

$$\xi^{a(t)} \equiv \left(x^{a(t)},\ y^{a(t)},\ z^{a(t)}\right) \tag{2.23}$$

to $\overline{\text{RFP}}^{a(t)}$, we could not possibly have $\bar{x}_i = \bar{y}_i \bar{z}_i,\ \forall i$, because then, from Lemma 1, $\xi^{a(t)}$ would be feasible to $\text{RFP}^{a(t)}$, thereby yielding $UB_{a(t)} \le \nu^*$, which is a contradiction. Therefore, we can find a branching variable and partition the node subproblem based on the proposed branching rule, and proceed to the next stage. A formal statement of the branch-and-bound algorithm is given below.

Having formally designed a branch-and-bound algorithm for solving Problem RFP, some remarks are now in order:

(a) Note that, during the course of the branch-and-bound algorithm, there are other potential choices for selecting a branching variable. For example, given an $i^* \in \arg\max_{i \in V} \left|\bar{x}_i - \bar{y}_i \bar{z}_i\right|$, we could have imposed the dichotomy on the y-variables as: $y_{i^*} \le \left\lfloor \bar{y}_{i^*} \right\rfloor$ and $y_{i^*} \ge \left\lfloor \bar{y}_{i^*} \right\rfloor + 1$, which in turn could lead to additional restrictions on the *z*-variables based on the following dichotomies, corresponding to the left and right children nodes, respectively as:

$$\frac{1}{\left\lfloor \bar{y}_{i^*} \right\rfloor} \le z_{i^*} \le 1 \quad \text{and} \quad \frac{1}{1+\left|N(i)\right|} \le z_{i^*} \le \frac{1}{1+\left\lfloor \bar{y}_{i^*} \right\rfloor} .$$

(b) Next, as elucidated in Balas and Yu (1986), an oft repeated criticism of applying branch-and-bound approaches to solving Problem MIS is the inherent asymmetry that is created in the search tree by fixing the x-variables to either one (which in turn results in fixing all the neighborhood variables to zero) or zero (which has no impact on the other variables). In our case, we cannot enforce such variable fixing strategies in Problem RFP (as this would eliminate the clique solutions), our computations reveal that the branch-and-bound tree is not consigned to an asymmetric growth pattern, and is of a more balanced structure.

(c) Furthermore, the various lower and upper bounding mechanisms described in this paper can be gainfully utilized to significantly curtail the size the search tree.

Table 10: Branch-and-bound algorithm for solving Problem FP

Algorithm 9. Branch-and-Bound Algorithm for Problem RFP

Step 0: (*Initialization*) Set $k=0$ and $k_{\max}=0$. Define $L=\{k\}$, and the current incumbent solution (obtained by the heuristic) as $v^*=LB$. Go to Step1.

Step 1: (*Node Selection Step*) If $L=\varnothing$, stop; return v^* as optimum. Else, select an active node from L, say node k, and solve RFP^k. Let $v\left[\text{RFP}^k\right]$ be the optimum objective function value at node k. Go to Step 2.

Step 2: (*Fathoming Step*) If $v\left[\text{RFP}^k\right]\le v^*+1$, then fathom node k, and update $L=L\setminus\{k\}$, and go to Step 1. If $v\left[\text{RFP}^k\right]\ge v^*+1$ and feasible to RFP, then set $v^*=v\left[\text{RFP}^k\right]$, fathom node k, update $L=L\setminus\{k\}$, and go to Step 1. Else, go to Step 3.

Step 3: (*Branching Step*) Create child nodes $k_{\max}+1$ and $k_{\max}+2$, and update $k_{\max}=k_{\max}+2$. Set $L=L\cup\{k_{\max}+1,\ k_{\max}+2\}$. Go to Step 1.

2.4 Computational Experience

In this section, we report some preliminary computational results based on standard graph data sets obtained from Balasundaram and Butenko (2005). All of our computations are conducted on a Windows 7 machine, equipped with an Intel Core i7-2640M 2.80GHz processor and 6GB RAM, utilizing MATLAB R2010b in conjunction with GuRoBi 6.0.4 (www.gurobi.com) as the underlying LP/IP solver.

Note that there are certain types of graphs that prove to be difficult test instances for the various algorithms presented in this paper, and the level of difficulty is dependent on the *density* of the graph. Recalling that for simple, undirected graphs, the graph density $\rho=\dfrac{2|E|}{n(n-1)}$, we sampled several graphs from the DIMACS website, and plotted the gap between the lower bound (obtained by the simple greedy heuristic, namely Algorithm 7) and the upper bound (obtained by solving the LP relaxation of Problem MIS) as a function of graph density. As seen in Figure 6, the UB-LB gap is small for low density and high density graphs, which is to be expected as for graphs with low/high density, the greedy algorithm obtains a near optimal solution, and the upper bound is fairly tight (owing to the maximal clique inequalities, etc.) as well. Moreover, if the UB-LB gap is small, then a branch-and-bound (or any other search) algorithm can close the gap very efficiently, and such instances do not really yield adequate insights. However, graphs with $0.2 \leq \rho \leq 0.45$ pose a greater challenge, and hence, we handpicked graphs that are in this density range to test the relative performances of the various algorithms in our study.

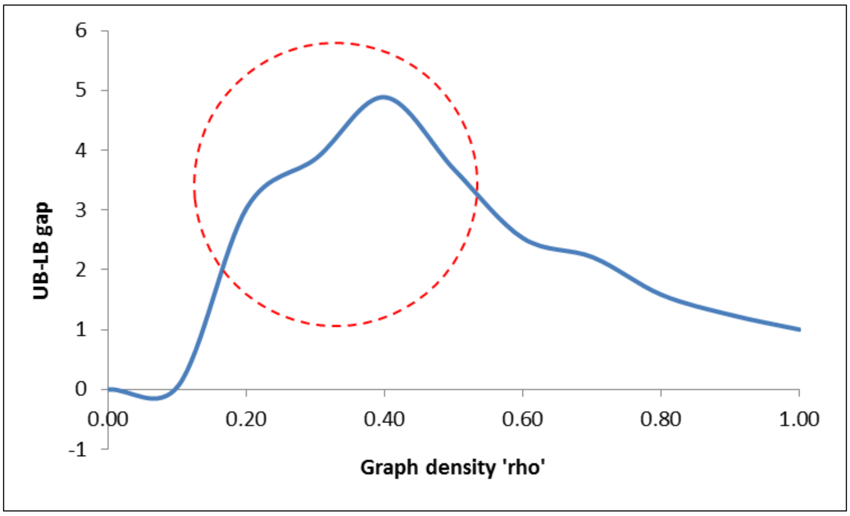


Figure 6. The upper-lower bound gap for Problem MIS at the root node as a function of graph density.

In our computations, we begin by comparing the lower bounds on the independence number obtained by six different methods: Caro-Wei's inequality (2.18); the degree-based greedy heuristic (see Table 8); the LP-based rounding heuristic (see Table 9); the maximal independent sets obtained by exploiting the polynomial programming formulations (2.7a) and (2.7b) using Algorithm 1 (see Table 1) and Algorithm 2 (see Table 2), respectively; and finally, the maximal independent set derived from the vertex sets in Algorithm 5 (see Table 5).

Tables 11 and 12 list the lower bounds on the independence number, along with the consumed CPU times (in seconds), obtained by these various algorithms. As seen from the recorded results in Table 11, on average, the heuristic procedure consumed 0.0037 seconds as compared to 0.007 seconds taken by the LP-based rounding heuristic, and the Caro-Wei's lower bound required 0.0093 seconds. However, the LP rounding heuristic has the least percentage deviation from the optimum, given by 11.46%, followed by the degree-based heuristic at 21.21%, and finally the Caro-Wei's lower bound is a distant third with an average deviation of 42.24% from the optimum.

Table 11: Comparisons of lower bounds on independence number: Part-I

Graph	|V|	|E|	α(G)	Caro-Wei LB		GH		LP-based RH	
				LB	CPU (s)	LB	CPU (s)	LB	CPU (s)
g7-1	7	16	3	2	0.0091	2	0.0027	2	0.0041
g10-1	10	13	5	3	0.0077	4	0.0017	5	0.0072
g10-2	10	22	4	3	0.0090	3	0.0014	3	0.0037
g15-1	15	45	5	3	0.0096	4	0.0028	4	0.0351
g15-2	15	19	9	5	0.0126	7	0.0033	9	0.0020
san15-1	15	35	6	3	0.0102	5	0.0024	5	0.0081
san15-2	15	20	8	5	0.0099	6	0.0026	8	0.0021
san30-1	15	43	15	9	0.0091	12	0.0049	15	0.0023
san40-1	40	78	20	10	0.0090	16	0.0066	20	0.0026
1dc.64	64	543	10	4	0.0083	8	0.0066	8	0.0070
1et.64	64	264	18	10	0.0081	16	0.0066	16	0.0103

In the second set of lower bound comparisons, we record the lower bounds on the independence number obtained from the different maximal independent set generating algorithms, namely Algorithms 2 and 3 (from Abello *et al*. (2001)) and Algorithm 6 (prescribed in this paper). Recognizing that the starting (initial) solution is critical to the success of these methods, in the results reported in Table 12, five randomly generated starting points were selected for each method, and the best result is displayed. As seen from Table 12, Algorithm 6 performs the best by producing independent sets with objective function values closest to the independence number with an average deviation of only 13.1%, whereas Algorithms 2 and 3 produce average deviations of 25.25% and 15.95%, respectively. However, Algorithm 3 is relatively the quickest, consuming only 0.0078 seconds on average as compared to 0.0112 seconds and 0.025 seconds by Algorithms 2 and 6, respectively. A closer observation reveals that Algorithm 3 outputs a better solution when compared to Algorithm 6 in only one instance (Problem san40-1), and in all other instances the maximal independent sets generated by the underlying vertex sets are significantly superior. Indeed, even for the most challenging of our instances, namely Problem 1et.64, Algorithm 6 generates a very strong lower bound, whereas the other maximal independent set generating algorithms are unable to avoid the trap of a strong local optimum with a lower bound value of 8.

Table 12: Comparisons of lower bounds on independence number: Part-II

Graph	|V|	|E|	α(G)	Algorithm 2		Algorithm 3		Algorithm 6	
				LB	CPU (s)	LB	CPU (s)	LB	CPU(s)
g7-1	7	16	3	2	0.011	3	0.005	3	0.0093
g10-1	10	13	5	4	0.012	4	0.005	4	0.0063
g10-2	10	22	4	3	0.01	3	0.005	3	0.0035
g15-1	15	45	5	4	0.013	4	0.005	5	0.0091
g15-2	15	19	9	7	0.01	9	0.005	7	0.0169
san15-1	15	35	6	5	0.02	5	0.005	5	0.0095
san15-2	15	20	8	6	0.01	8	0.005	7	0.0182
san30-1	15	43	15	12	0.02	13	0.005	14	0.0485
san40-1	40	78	20	16	0.005	19	0.015	16	0.0627
1dc.64	64	543	10	8	0.0057	8	0.015	9	0.0259
1et.64	64	264	18	8	0.0067	8	0.016	16	0.0687

Comparing across Tables 11 and 12, it is evident that the LP rounding-based heuristic as well as the maximal independent set algorithm based on vertex sets defined in this paper perform admirably in determining tight lower bounds on the independence number. Moreover, despite its simplicity, the degree-based greedy heuristic is a quick and viable alternative to determining good quality independent sets. More importantly, all of these methods, with the exception of the Caro-Wei lower bound, also yield *feasible* (and possibly incumbent) solutions to both Problems MIS and FP, and the relative merits of each of these bounding methods as measured by the quality of solution and computational resources expended is evident from these computational experiments.

Next, akin to comparing lower bounds, we also compared the upper bounds on the independence number obtained by three different methods, namely the optimum objective function values of Problem $\overline{\text{MIS}}$, Problem $\overline{\text{RFP}}$, and the convex program (2.21) solved using the global optimization solver OQNLP (http://tomwiki.com/OQNLP). As seen in Table 13, the average CPU time taken by Problem $\overline{\text{MIS}}$ (= 0.175) is lesser than that consumed by Problem $\overline{\text{RFP}}$ (= 0.348), which in turn is significantly lower than CP-OQNLP (= 1.965). Conversely, the average percentage deviation of the optimum objective function value of the convex program from the independence number (= 17.47%) is significantly lesser than that for Problem $\overline{\text{MIS}}$ (= 39.80%), which is closely followed by Problem $\overline{\text{RFP}}$ (= 40.77%). Clearly, in this case, the additional computational effort expended by the convex program does indeed help in determining a far improved upper bound.

Table 13: Comparisons of upper bounds on independence number

Graph	|V|	|E|	α(G)	IP-relaxed		RFP-relaxed		CP-OQNLP	
				UB	CPU (s)	UB	CPU (s)	UB	CPU (s)
g7-1	7	16	3	3	0.192	3	0.124	3	2.304
g10-1	10	13	5	5	0.089	5	0.108	5	1.499
g10-2	10	22	4	5	0.109	5	0.105	4	0.883
g15-1	15	45	5	7	0.101	7	0.107	6	2.805
g15-2	15	19	9	9	0.086	9	0.104	9	2.098
san15-1	15	35	6	7	0.101	7	0.106	7	1.835
san15-2	15	20	8	8	0.09	8	0.108	8	2.042
san30-1	30	43	15	15	0.104	15	0.108	17	2.009
san40-1	40	78	20	20	0.997	21	2.812	24	2.143
1dc.64	64	543	10	32	0.035	32	0.036	16	1.9
1et.64	64	264	18	33	0.022	33	0.11	22	2.097

Finally, we implemented the branch-and-bound algorithm described in Section 2.3 for solving Problem RFP, and recorded the number of nodes enumerated as well as the CPU time (seconds) consumed in determining the optimal solution. A similar computational exercise was conducted on Problem MIS, which was directly solved using GuRoBi as the default solver engine. To provide a fair comparison, as existing large-scale commercial LP solvers, such as GuRoBi, can determine an optimal solution to a 0-1 MIP very efficiently if a *feasible solution* is provided as an input, in our implementation of the branch-and-bound algorithm, we follow the tailored branching strategy only until a *first feasible* solution is obtained, and subsequently, we follow GuRoBi's default branch-and-cut strategy until convergence to the optimum is obtained. From the results recorded in Table 14, it can be seen that Problem MIS outperforms Problem RFP, both in terms of the number of nodes enumerated as well as the CPU time taken to converge to the optimal solution. A similar observation holds true for the same performance metrics in finding the *first feasible* (FF) solution. Indeed, for the two largest instances, Problem RFP fails to converge within the prescribed time limit of 7200 seconds.

Table 14: Comparison of branch-and-bound algorithms on Problems MIS and RFP

Graph	|V|	|E|	α(G)	IP formulation					RFP formulation				
				#nodes	CPU (s)	#nodes (FF)	CPU (s) (FF)	Value (FF)	#nodes	CPU (s)	#nodes (FF)	CPU (s) (FF)	Value (FF)
g7-1	7	16	3	13	0.1165	7	0.0733	1	21	0.2373	5	0.0848	3
g10-1	10	13	5	3	0.0336	2	0.0238	5	9	0.1963	4	0.0424	5
g10-2	10	22	4	23	0.2107	8	0.0900	2	75	0.7560	7	0.0896	4
g15-1	15	45	5	57	0.5155	13	0.1313	2	659	17.5714	18	0.2070	4
g15-2	15	19	9	1	0.0144	1	0.0142	9	7	0.0926	3	0.0455	9
san15-1	15	35	6	31	0.2984	11	0.1244	4	935	10.0693	13	0.1550	5
san15-2	15	20	8	1	0.0175	1	0.0174	8	9	1.7856	10	0.1282	7
san30-1	30	43	15	9	0.0917	3	0.0384	15	41	23.2507	26	0.3509	12
san40-1	40	78	20	7	0.0768	4	0.0453	20	51	1156.27	38	0.5983	14
1dc.64	64	543	10	95159	3422.59	63	2.6472	2	-	-	826	8.3840	8
1et.64	64	264	18	320023	4104.20	58	0.7178	7	-	-	1438	23.991	8

Recognizing the computational difficulty in obtaining optimal solutions by exclusively using branching strategies, an improved version of the branch-and-bound algorithm was implemented, wherein the LP rounding-based heuristic (Algorithm 8) was utilized to derive a feasible solution at every node. This feasible solution was in turn (possibly) improved by running the newly designed maximal independent set algorithm based on vertex sets (Algorithm 5), and these feasible solutions were utilized to fathom nodes by bound in the search tree as well as update the incumbent solution (if possible). These improved comparisons are presented in Table 15, where, with these enhanced techniques, the optimal solution was determined in all cases corresponding to Problem RFP as well. In this case, the first feasible solution is always determined at the *root node* at very little computational expense. Moreover, in several instances, as the lower and upper bounds (determined by any one of the various aforementioned bounding schemes) are equal, the optimal solution is determined at the root node. Furthermore, while all of the performance metrics still favor Problem MIS over Problem RFP, this additional effort consumed by Problem RFP is as a result of the search tree trying to determine all of the alternate optima present in the problem, which is considerably greater for Problem RFP as seen in our characterization of alternate optima (refer Theorem 5 and its surrounding discussion).

Table 15: Improved comparisons of branch-and-bound algorithms on Problems MIS and RFP

Graph	|V|	|E|	α(G)	Problem MIS					Problem RFP				
				#nodes	CPU (s)	#nodes (FF)	CPU (s) (FF)	Value (FF)	#nodes	CPU (s)	#nodes (FF)	CPU (s) (FF)	Value (FF)
g7-1	7	16	3	3	0.0168	1	0.0041	2	4	0.1149	1	0.0107	3
g10-1	10	13	5	1	0.0063	1	0.0072	5	1	0.0603	1	0.0054	5
g10-2	10	22	4	9	0.0372	1	0.0037	3	9	0.2751	1	0.0250	4
g15-1	15	45	5	23	0.0762	1	0.0351	4	75	2.2041	1	0.0282	4
g15-2	15	19	9	1	0.0055	1	0.0020	9	1	0.0307	1	0.0055	9
san15-1	15	35	6	9	0.0397	1	0.0081	5	14	0.4391	1	0.0242	5
san15-2	15	20	8	1	0.0070	1	0.0021	8	1	0.2272	1	0.0060	8
san30-1	30	43	15	1	0.0044	1	0.0023	15	1	1.8646	1	0.0037	15
san40-1	40	78	20	1	0.0068	1	0.0026	20	4	6.0133	1	0.0061	20
1dc.64	64	543	10	65949	183.967	1	0.0070	8	285031	1767.69	1	0.2975	8
1et.64	64	264	18	69097	845.517	1	0.0103	16	616642	3004.52	1	0.1997	16

Indeed, in our final measurement, Table 16 reports the lower bounds on the number of alternate optima, obtained using Theorem 5 and Equation (2.12), as well as the exact number of alternate optima for both Problems MIS and FP. The maximum independent sets used to generate these lower bounds are also included in the table below. As an example, consider the largest graph in our test set, namely 1et.64, and in this case, the total CPU time consumed by Problem FP is nearly four times greater than that of Problem MIS, but the number of alternate optimal solutions for these problems follows nearly the same ratio. As such, based on these results, it can be seen that both Problems MIS and RFP consume similar *computational effort per optimal solution*, which validates the merits and relative strength of the continuous formulation in comparison to the integer programming approach.

Table 16: Lower bounds and exact number of alternate optimal solutions to Problems MIS and FP

Graph	|V|	|E|	α(G)	Generating Maximum Independent Set	Theorem 5		Equation (12)		Exact Number of Alternate Optima	
					MIS	FP	MIS	FP	MIS	FP
g7-1	7	16	3	[5 6 7]	1	1	1	1	1	1
g10-1	10	13	5	[2 3 8 9 10]	2	3	2	3	2	3
g10-2	10	22	4	[2 8 9 10]	1	1	1	1	1	1
g15-1	15	45	5	[1 3 6 8 9]	2	3	2	3	2	3
g15-2	15	19	9	[2 3 4 6 7 8 11 14 15]	1	1	1	1	1	1
san15-1	15	35	6	[6 7 9 10 11 12]	2	3	2	3	2	3
san15-2	15	20	8	[2 3 6 8 10 11 12 14]	1	1	1	1	1	1
san30-1	15	43	15	[1 3 5 7 9 14 15 17 18 20 21 27 28 29 30]	8	27	8	27	8	27
san40-1	40	78	20	[3 5 7 10 11 12 13 14 18 21 22 23 24 25 28 29 30 32 35 28]	8	27	8	27	8	27
1dc.64	64	543	10	[1 4 13 16 19 46 49 52 61 64]	1	1	1	1	1	1
1et.64	64	264	18	[1 2 4 8 9 11 15 16 25 29 30 32 36 50 52 57 60 64]	6	21	4	0	16	63

2.5 Conclusions and Extensions

In this chapter, we studied the NP-Hard maximum independent set problem from a continuous fractional programming formulation perspective (Problem FP), developed a global optimization algorithm to solve this problem, and showed that this continuous formulation stays on par with the well-known 0-1 linear programming formulation (Problem MIS) with respect to various performance metrics. In this context, we defined a new class of *vertex sets,* and utilized the structure of these vertex sets to derive explicit characterizations of the number of alternate optima present in both Problem MIS as well as Problem FP. Moreover, these vertex sets also enabled a simple, yet powerful, construction procedure to efficiently determine maximal independent sets. Similarly, we also presented a fast LP-rounding based heuristic to determine tight lower bounds on the independence number. Putting these theoretical contributions together, as seen in our numerical experiments, we showed that the computational time required per optimal solution is comparable and in some instances lower for Problem FP as compared to Problem MIS (as the number of alternate optima for Problem FP is significantly greater when compared to Problem MIS).

Delving deeper, these vertex sets and their resulting influence on the number of alternate optima to Problems MIS and FP results in an interesting conjecture. While we have not provided a formal proof of this result, this result has proved to be true in all of our computational experiments, and provides another viewpoint towards understanding the interconnections between vertex sets and independent sets. The conjecture states that, for connected graphs, if $\bar{x}_i = \frac{1}{2}, \ \forall i = 1, \ldots, n$, is an *optimal solution* to Problem $\overline{\text{MIS}}$, then the refined vertex sets only yield a lower bound on the number of alternate optima to Problems MIS and FP, respectively. (Note that this property is usually true for highly symmetric graphs, which are often exceptions to computing the exact number of alternate optimal solutions.) A formal statement of this conjecture is given next.

CONJECTURE 1. *If* $\bar{x}_i = \frac{1}{2}$, $\forall\, i = 1,\ldots,n$, *is an optimal solution to Problem* $\overline{\text{MIS}}$ *over a connected graph G, then for any maximum independent set* I^*, *we have* $\Lambda_{\text{MIS}} > \prod_{k\in I^*}\left|\tilde{C}^{(k)}\right|$ *and* $\Lambda_{\text{FP}} > \prod_{k\in I^*}\left|\tilde{C}^{(k)}\right|$, *where* $\tilde{C}^{(k)}$ *denote the refined (disjoint) vertex sets.* ■

COROLLARY 3. *If* $\bar{x}_i = \frac{1}{2}$, $\forall\, i = 1,\ldots,n$, *is not an optimal solution to Problem* $\overline{\text{MIS}}$ *on a connected graph G, and* $C^{(k)} \equiv \tilde{C}^{(k)}$, $\forall\, k \in I^*$, *i.e., if* $\tilde{E}^0 \equiv \varnothing$, *then the bounds obtained in Theorem 5 are tight.* ■

Another interesting (unproven) conjecture that follows from these vertex sets related to maximal cliques is given below, and we are currently investigating these conjectures and related insights are forthcoming.

CONJECTURE 2. *Define* $C_{\max} = \max_{k\in I^*}\left\{\left|C^{(k)}\right|\right\}$. *Then, there exists a maximal clique in a connected graph G of cardinality* $C_{\max}$.

Chapter 3

A Discrete Optimization Framework for DAS Locations in CDMA Cellular Networks

In this chapter, we present a set of (exact and approximate) mathematical models and algorithms for determining the set of (globally) optimal distributed antenna deployments and the supported user demand in cellular code division multiple access (CDMA) systems. We focus on the uplink (user-to-base station) formulation and assume that the base station combines all the received signals at each of the antennas using path-gain based weights. In CDMA systems, as all users occupy the system bandwidth at the same time thereby interfering with each other, this results in a complicated mixed-integer 0-1 multilinear programming problem, where the objective function maximizes the total system capacity, while ensuring that the minimum signal-to-interference-plus-noise-ratio (SINR) constrains and maximum transit power constraints for each user are satisfied. The highly nonlinear, nonconvex problem is reformulated to yield a tight mixed-integer 0-1 linear programming representation via the addition of several auxiliary variables and constraints, and a specialized algorithm is designed to determine globally optimal solutions. The computational result obtained, corresponding to different user distributions, demonstrate the efficacy of the proposed models and algorithms, and clearly exhibit the viability of both exact and approximate models.

This chapter is organized as follows. In Section 3.1, we first present a detailed introduction of the cellular network and the CDMA system. In Section 3.2, the underlying assumptions required to obtain a clear definition of the problem are stated, and subsequently, in Section 3.3, we derive various exact and approximate mathematical formulations, and prescribe tailored algorithmic methodologies for determining globally optimal solutions for the problem under consideration. Detailed computational experience related to each model formulation is presented in Section 3.4, and finally, Section 3.5 discusses conclusions as well as extensions for future research.

3.1 Introduction

In typical cellular networks, a hand-held terminal, referred to as a *user*, employs a single-antenna ratio to communicate with a local *base station*. The base station comprises $L > 1$ sufficiently spaced *antennas elements* (AEs) that transmit and receive signals with a user located within its *coverage* (or *service*) area called a *cell*. Because wireless signals use multiple paths when propagating from a transmitter to a receiver, the signal received at an antenna is a superposition of multiple copies of the transmitted signal. This results in either constructive or destructive interference, which amplifies or attenuates the power of the received signal. Strong destructive interference, *i.e.*, a deep fade, is extremely undesirable as it results in a (temporary) failure of communication. Having $L > 1$ sufficiently spaced base station antennas results in L independent sets of fading paths between a transmitter and receiver, thereby reducing the overall probability of a deep fade from p (for a single transmit-receive antenna pair), where $0 < p < 1$, to the order of p^L. Thus, the use of L base station antenna helps protect transmissions made between the user and the base station against multipath fading, and this reliability improvement is referred to as the *diversity gain* of applying multiple antennas in wireless communications.

Distributed antenna systems (**DAS**) provide an alternate arrangement to achieve improved diversity gains in cellular networks by simultaneously increasing the overall capacity and coverage performance. In the DAS framework, the L antenna elements of a base station are deployed throughout the cell, as opposed to being installed at a common ratio tower, and are connected to the base station using dedicated links, *e.g.*, fiber optic, coax, or radio connections. An immediate advantage of distributing antennas (over co-locating them at a tower) is improved coverage with the cell, and can even lead to larger cells. Furthermore, by shortening the distance between a user and the antenna tower (*the access distance*), distributed antenna systems lead to a significant reduction in the required transmit power, which in turn reduces co-channel interference and results in improved capacity. Moreover, these L antennas located at a common base station tower, provided that the antenna elements are placed with sufficient inter-antenna spacing.

Several recent studies have demonstrated the aforementioned benefits of DAS deployment in cellular networks (see Amaldi *et al.*, 2001, 2003, 2006, Dai, 2008, Dai *et al.*, 2005, Hu *et al*., 2007, Obaid and Yanikomeroglu, 2000, Roh *et al*., 2002, Saleh *et al*., 1987, Yanikomeroglu and Sousa, 1993, Zhang *et al*., 2008, and Zhuang *et al*., 2003). To achieve optimal capacity gains using a DAS, the antenna locations must be carefully chosen such that the transmit powers are minimized, *i.e.*, interference is minimized and capacity is maximized. As demonstrated in Dai (2008), with uniformly distributed user and 16 regularly spaced antennas within a cell, the sum transmit power of a DAS is less than 5% of that consumed by conventional ci-located antennas. This tremendous power improvement results in part due to the reduced access distance and because the chosen antenna locations closely "match" the user distribution. However, for more general, non-uniform cases, optimal antenna locations based on the geography of user demand are required to effectively harness the capacity gain of the DAS architecture.

The combined base station location and user assignment problem for *Cellular Code Division Multiple Access* (**CMDA**) networks has been studied in several papers including Amaldi *et al*. (2001, 2003, and 2006), St-Hilaire, *et al*. (2006), *etc*. In Amaldi *et al*. (2003), the model presented minimizes the total cost of antenna deployments and system interference whiling ensuring that the system has enough capacity to allow all user simultaneously access to the network. Conversely, in Mathar and Schmeink (2001), the system capacity (defined as the number of users supported by the base station) is maximized subject to a budget constraint. Other models related to profit maximization, typically computed as the difference between revenue generated and the cost if locating a base station antenna per user, can be found in Galota *et al*. (2001) and Kalvenes *et al*. (2006). Of recent interest are also papers that maximize revenue (or capacity) in the presence of demand uncertainty in CMDA networks (see Laguna, 1998, Olinick and Rosenberger, 2008, Riis and Lodahl, 2002, Rosenberger and Olinick, 2007, Sen *et al*., 1994, and Sohn and Jo, 2006).

Note that the aforementioned papers examine the optimal locations of base stations with cellular CDMA systems while accounting for inter-user interference. Although, the problem of installing several (single-antenna) base stations with a network appears similar to installing several distributed antennas, there is one inherent difference. Specifically, in the former case the base stations do not combine user signals except for those located at the boundary between two cells (referred to as “soft-handoff”). However, in the DAS approach, the base station combines the L copies of each in-cell user’s signal. This not only alters the problem and solution but enables us to consider the approximate approach proposed in the sequel.

Another important aspect of current cellular networks is the *handover* or *handoff* process. A handover process occurs when a user’s signal is transferred from one base station to another. This could occur due to several reasons. The most common one is when a user moves out of the coverage area of the original base station and into the coverage area of a neighboring base station. Another common occurrence is when the capacity for connecting or making new calls within a cell has reached its peak capacity. In this case, some users who are located in a region with coverage from multiple cells would experience a handover to a cell with available capacity. This allows the cell operating at full capacity to provide service to users located in a region exclusively covered by that cell. In a CDMA network, a handover could also be done if there is a smaller neighboring cell in order to reduce the interference cause by the near-far effect.

There are two types of handover processes, namely the hard handover process and the soft handover process. In the hard handover process, the connection between the user and the original base station is terminated before the connection to the next base station is initiated (Laguna, 1998). This handover process is almost instantaneous and works so as to provide the least disruption to the user. In some instances where the user is in a region between two cells, the connection may switch back and forth between the cells depending on which connection is better. Conversely, in the soft handover process, a connection between the new base station and the user is established before the original connection is terminated. The time period where the user is simultaneously connected to two (or more) base stations in parallel is kept as short as possible to reduce the hogging of available bandwidth. An advantage of using this method is that the probability of disruption to the user is very low as a new connection is established before the original connection is terminated. However, the disadvantage of using this method is that while a user is connected in parallel to several base stations, several channels in the network are used to support a single user resulting in reduced capacity in the network.

Hence, an analytical framework that provides a way of computing optimal antenna deployments so that the total capacity is maximized in a DAS setting would be critical for service providers considering the DAS approach. To the best of our knowledge, no such framework currently exists. In this chapter, we pose and solve the problem of determining optimal *distributed antenna* deployments for maximizing the total number of admitted users in a CMDA system.

3.2 Problem Definition

We consider a region $\Re$ where the location of one base station is known and an optimal location for a set of distributed antennas is desired. Our goal is to determine these locations so that the uplink user capacity, *i.e.*, the number of simultaneous, single-rate, uplink users, is maximized. For ease of discussion, we refer to a distributed antenna as a *DAS antenna* and describe the location of each DAS antenna within $\Re$ using Cartesian coordinates. In this article, we treat the given base station and each DAS antenna as single-antenna systems. (The model presented in this chapter can be easily extended to the case of multiple base station and/or multiple antenna base stations.) Within the region $\Re$, we model traffic requests using a set of *demand nodes* , indexed by $i \in \{1,...,I\}$. Since user demand is typically a continuous function over the network's geographic region, these demand nodes represent a discretized version of the connection requests within the region. For example, the entire geographic region can be subdivided into I smaller subregion, located at its center of gravity. For each demand node i, let $(\alpha_i, \beta_i) \in \Re$ denote the location of demand node i, and let h_i denote the number of desired connections, *i.e.*, user terminals, from node i. This approach of modeling user demand via *discrete demand nodes*, referred to as "hotspots", has been advovated in several earlier studies on cellular network planning. (Amaldi *et al*., 2003))

To simplify discussion, let $h_i \equiv 1$, $\forall\, i = 1, \ldots, I$. Note that cases of non-unit demand can be easily incorporated by placing multiple demand nodes at, or in the vicinity of, a particular location. In typical CDMA networks, as demand nodes are admitted to a cell on a user-by-user basis, *i.e.*, the base station admits individual users from a demand node one-by-one rather than all $h_i > 1$ at once (*i.e.*, an all-or-nothing acceptance policy), modeling a demand node with $h_i > 1$ (essentially a "hotspot") as several identical nodes, each with demand $h_i \equiv 1$, is a reasonable and viable approach to account for such an admission policy.

REMARK 1. Here, we assume that: (1) at each base, ideal RAKE processing is performed on each user's received signal in order to maximize diversity; and (2) each user-base path is "infinitely dispersive", implying that there is a near-infinitude of resolvable, comparable-strength multi-paths on each link. Given (1) and (2), the receiver output signal sample for each user will be non-fading just as if the link had a single path of fixed gain. We assume these fixed gains are known and designate the path gains from demand node i to the given base as $\tilde{T}_i$ and from demand node i to a DAS antenna located at position (x, y) as $T_i(x, y)$. With these fixed gains, we can proceed by assuming the *additive white Gaussian noise* (AWGN) condition for each link.

Ideally, we would like to determine the optimal locations for the DAS antennas over the continuous range of points in $\Re$. However, the difficulty with this goal lies in expressing $T_i(x, y)$ as a *continuous function* over the range of x and y, as $T_i(x, y)$ depends on several factors such as the distance between demand node i and DAS antenna at location (x, y), the attenuation due to shadowing, which in turn is related to the nature of the obstacles between the demand node and the DAS antenna. (See Rappaport (2002) for a more detailed description of factors affecting path gains from demand node to DAS antennas.) More realistically, network designers are often limited by local ordinances and regulatory constraints, which limit possible antenna deployments to a *discrete set of candidate antenna* *locations*. Moreover, field measurements of $T_i(x, y)$ are available to network designers for a limited set of candidate locations. Therefore, instead of soling the continuous location problem, we focus on finding the optimal DAS antenna locations from amongst a discrete set of points.

Accordingly, let the set of candidate DAS antenna locations be indexed by $k \in \{1,...,K\}$ where each location k has Cartesian coordinates $(x_k, y_k) \in \Re$. We let $k = 0$ denote a given base station located at the origin. For notational ease, we will henceforth refer to $T_i(x_k, y_k)$ as T_{ki}, $\forall(i, k)$, and additionally, we note that $\tilde{T}_i = T_{0i}$, $\forall i = i,...I$. Given these definition, the *signal gain* between each user at demand node i and the given base can expressed as $\sqrt{T_{0i}}e^{j\phi_{0i}}$. Similarly, the signal between this user and an antenna deployed at location (x_k, y_k), $k = 1,...K$, can be written as $\sqrt{T_{ki}}e^{j\phi_{ki}}$. The parameters ϕ_{0i} and ϕ_{ki} are the phases of the signal gains from user *I* to the given base and a DAS antenna at location k, respectively. In the DAS system studies here, *all* antenna receive, dispread, and RAKE-combine the signal from each admitted. Therefore, there are *multiple* output streams (one at the given base and one at each of the distributed antennas), which contain a desired use's signal. Assuming that all admitted user terminals use single-rate CMDA over the same set of frequencies, each of these streams also contains interference from the supported users plus thermal noise. We assume the output streams at the given base and DAS antenna at location k are weighted by ω_{0i} and ω_{ki}, respectively, and then summed over all antennas.

REMARK 2. Throughout this chapter, we assume that the chosen weight ω_{0i} and ω_{ki}, $\forall k = 1,...K$, reflect the relative path gains between users and system antennas, *i.e.*, an antenna with a higher path gain to an admitted user has a proportionately weight than an antenna with a lower path gain to this user. Details of this weighting scheme are provides in the discussion below.

Then, the object of the *DAS Location Problem* (**DASLP**) under consideration is to determine an optimal set of $L(\leq K)$ locations (chosen from the set of candidate locations within $\Re$ where DAS antennas maybe deployed), such that the total number of user terminal supported is *maximized* subject to given *quality of service* (QoS) requirements. These QoS requirements are expressed in terms of minimum *Signal-to-Interference plus Noise Ratio* (**SINR**) constraints on the uplink. (We model QoS requirements as uplink SINR constraints because in many applications they tend to be more stringent than the corresponding downlink SINR constraints (refer Sherali and Tuncbilek, 1992).) In doing so, the solution to the DASLP also determines the optimal subset of the I demand node that can be admitted into the overall system. Hence, a solution to the DASLP: (1) dictates how the DAS system should be optimally deployed, and (2) determines how much of the (user) demand can be supported by the resulting system.

3.3 Model Formulation and Algorithms

3.3.1 An Exact System Model

As aforementioned, we consider the DAS antenna location problem for each admitted user is processed by all $L+1$ system, regardless of the user's path gains to these antennas and furthermore, we also assumed that all antennas receive, dispread, and RAKE-combine the signal from admitted user. These output streams are then weighted and summed, resulting in the *overall output SINR* for a user at demand node i. To formulate this uplink SINR term, determine

$$z_i = \begin{cases} 1, & \text{if demand node } i \text{ is admitted into the system} \\ 0, & \text{otherwise} \end{cases}, \tag{3.1}$$

$$q_i = \begin{cases} 1, & \text{if DAS antenna candidate location } k \text{ is selected} \\ 0, & \text{otherwise} \end{cases}, \tag{3.2}$$

Then, the overall output SINR for a user at demand node i is given by,

$$\mathrm{SINR}_i = \frac{z_i \frac{W}{R} \left| \omega_{0i} \sqrt{P_i T_{0i}} e^{j\phi_{0i}} + \sum_{k=1}^{K} q_k \omega_{ki} \sqrt{P_i T_{ki}} e^{j\phi_{ki}} \right|^2}{\left|\omega_{0i}\right|^2 \left(\sum_{\substack{l=1 \\ l \neq i}}^{I} z_l P_l T_{0l} + \eta W \right) + \sum_{k=1}^{K} q_k \left|\omega_{ki}\right|^2 \left(\sum_{\substack{l=1 \\ l \neq i}}^{I} z_l P_l T_{kl} + \eta W \right)}, \tag{3.3}$$

where, P_i is the transmit power of user i (which is controlled by the given base), W is the system bandwidth, R is the supported rate of each admitted user, W/R is the processing gain, and ηW is the power of the AWGN at the receiver.

REMARK 3. The SINR equation (3.3) implicitly treats the interference from an admitted user at demand node i as uncorrelated between antennas, whereas in fact, the $L+1$ interferences are indeed correlated because they carry the same data. Therefore, the combined output voltage sample from an admitted user at demand node i is proportional to $\omega_{0i} \sqrt{P_i T_{0i}} e^{j\phi_{0i}} + \sum_{k=1}^{K} q_k \omega_{ki} \sqrt{P_i T_{ki}} e^{j\phi_{ki}}$, and the squared magnitude has a cross-term proportional to the cosine of a path-gain-related phase. This phase term is a random variable that is uniformly distributed on $[0, 2\pi)$, and so, the summation of all cross-term over the index $l \neq i$ converges to a mean of zero as the total number of users admitted is significantly greater than the number of antennas $\left(\sum_{i=1}^{I} h_i z_i >> L \right)$. Therefore, for the capacity levels of interest here, substitution of the mean value of zero is a very reasonable approximation; hence, (3.3) applies. Note that we have implicitly assumed that the number of active users will be much greater than the number of antennas. For this to hold, we are assuming that the system processing gain (W/R) is adequately strong; thus our focus is not on the number of high-rate users but rather on the number of moderate to low-rate users that can be simultaneously supported within the cell. The guiding design principle that we have followed here is that low-to-moderate rate connectivity to a larger user base is more desirable than high-rate connectivity to a select number of users.

The DAS system studied here uses path-gain-weighted combining, *i.e.*, it co-phase the $L+1$ received signals via the phase of ω_{0i} and ω_{ki} (assuming that a DAS antenna is present at location k) and use the following magnitudes for ω_{0i} and ω_{ki} :

$$|\omega_{0i}| = \frac{\sqrt{T_{0i}}}{\sqrt{T_{0i} + \sum_{k=1}^{K} q_k T_{ki}}} \quad \text{and} \quad |\omega_{ki}| = \frac{\sqrt{T_{ki}}}{\sqrt{T_{0i} + \sum_{m=1}^{K} q_m T_{mi}}}, \quad \forall (i,k). \tag{3.4}$$

Assuming these values and co-phasing, we can simplify (3.3) to obtain:

$$\mathrm{SINR}_i = \frac{z_i \left(\frac{W}{R}\right)\left(\frac{P_i}{\eta W}\right)\left(T_{0i} + \sum_{k=1}^{K} q_k T_{ki}\right)}{\sum_{\substack{l=1 \\ l \neq i}}^{I} \left[\frac{\left(T_{0i} T_{0l} z_l + \sum_{m=1}^{K} q_m T_{mi} T_{ml} z_l\right)\left(\frac{P_l}{\eta W}\right)}{T_{0i} + \sum_{k=1}^{K} q_k T_{ki}} \right] + 1}. \tag{3.5}$$

We are now ready to formulate the DASLP, where for notational convince, we have substituted $S_i \equiv \frac{P_i}{\eta W}, \quad \forall i = 1, \ldots, I.$

DASLP: Maximize $\sum_{i=1}^{I} z_i$ (3.6a)

Subject to $\sum_{k=1}^{K} q_k = L$, (3.6b)

$$K' T_{0i} z_i S_i \left(T_{0i} + \sum_{k=1}^{K} q_k T_{ki}\right) + K' \sum_{k=1}^{K} T_{ki} z_i S_i q_k \left(T_{0i} + \sum_{m=1}^{K} q_m T_{mi}\right) \geq$$
$$\sum_{\substack{l=1 \\ l \neq i}}^{I} T_{oi} T_{0l} z_i z_l S_l + \sum_{\substack{l=1 \\ l \neq j}}^{I} \sum_{m=1}^{K} T_{mi} T_{ml} z_i z_l S_l q_m + z_i \left(T_{0i} + \sum_{k=1}^{K} q_k T_{ki}\right), \quad \forall i \tag{3.6c}$$

$$S_i \geq z_i, \quad \forall i \tag{3.6d}$$

$$z_i, q_k \in \{0,1\}, \quad \forall (i,k), \text{ and } 0 \leq S_i \leq \mathrm{SNR}_{\max}, \quad \forall i \;. \tag{3.6e}$$

In this formulation, the objective function (3.6a) seeks to maximize the total number of admitted users, constraint (3.6b) restricts the number of DAS antenna locations to be a specified number *L*, and constraint (3.6c) imposes the QoS requirement that all admitted users must meet a minimum SINR of Γ, *i.e.*, $\text{SINR}_i \geq z_i\Gamma$, where $K' = W/R\Gamma$ such that $K'+1$ is the single-cell pole capacity of the given CDMA system. (The pole capacity is defined as the maximum capacity that can be supported in the cell with random CDMA user codes, negligible noise power, and perfect power control so that all users have a minimum SINR value of Γ. Hence, this will be the maximum (theoretically) supported capacity in the system and any solution to DASLP will be upper bounded by $K'+1$.) Constraint (3.6d) ensures that the power of each admitted user must be at least the power of the AWGN at the receiver, and finally constraint (3.6e) explicitly states the binary nature and lower/upper bounds on the (z, q)- and *S*-variables, respectively. Note that, although the objective function (3.6a) and the constraints (3.6b) and (3.6d) are linear, constraint (3.6c) is a *quartic multilinear polynomial term* in the variables (z, S, q). More importantly, the presence of binary variables and the multilinear terms in constraint (3.6c) renders DASLP as a nonconvex programming problem.

3.3.2 A Centrally Located Antenna Problem (CLAP)

Note that the SINR term (3.3), which restricts interference within the cell, can be readily modified to accommodate the scenario where all L deployed antennas are *co-located* at the base station, which is essentially the “baseline” case to be used for comparisons with the DAS approach. Assuming that the L antennas are deployed (typically as an array) at the base station and sufficiently spaced to ensure uncorrelated fading (as is the desirable case) we can rewrite the SINR in this case as follows:

$$\text{SINR}_i = \frac{z_i\left(\frac{W}{R}\right)\left(\frac{P_i}{\eta W}\right)\left(\sum_{j=1}^{L} T_{ji}\right)}{\sum_{\substack{l=1\\l\neq i}}^{I}\left[\frac{\left(\sum_{j=1}^{L} T_{jl}^2 z_l\right)\left(\frac{P_l}{\eta W}\right)}{\sum_{j=1}^{L} T_{ji}}\right] + 1}. \tag{3.7}$$

Here, we have removed the q_k-variables since all L antennas are located at the base station. Moreover, we still assume that T_{ji} is the path gain from demand node i to antenna j located at the base station, and furthermore, we assume the base station employs path-gain based combining across the L antennas. In this scenario, the necessary steps to compute the maximum capacity supported by this system would first involve setting a minimum SINR threshold for the cell, *i.e.*, $\text{SINR}_i \geq z_i \Gamma$, for all admitted users i. Thus, this *centrally located antenna problem* (**CLAP**) can be expressed as:

$$\textbf{CLAP:}\ \text{Maximize}\quad \sum_{i=1}^{I} z_i \tag{3.8a}$$

$$\text{subject to:}\quad K' z_i S_i\left(\sum_{j=1}^{L} T_{ji}^2\right) \geq \sum_{\substack{l=1\\l\neq i}}^{I}\sum_{j=1}^{L} T_{jl}^2 z_i z_l S_l + z_i\left(\sum_{j=1}^{L} T_{ji}\right) \tag{3.8b}$$

$$S_i \geq z_i, \qquad \forall\, i \tag{3.8c}$$

$$z_i \in \{0,1\},\ \ \forall (i,k), \text{ and }\ 0 \leq S_i \leq SNR_{\max},\ \ \forall i \tag{3.8d}$$

As is the case in DASLP above, in this formulation, the objective function (3.8a) seeks to maximize the total number of admitted users, constraint (3.8b) imposes the QoS requirement that all admitted users must meet a minimum SINR threshold, constraint (3.8c) ensures that the power of each admitted user must be at least the power of the AWGN at the receiver, and finally constraint (3.8d) explicitly states the binary nature and lower/upper bounds on the z- and S-variables, respectively. As aforementioned, the maximum supported capacity of this centrally located antenna problem will also be upper bounded by $K'+1$.

In our computations, we will compare the capacity that can be obtained, *i.e.*, the number of users who can be supported by the system, for both distributed and centralized models. Moreover, the algorithmic approach that we have adopted in this paper (see Section 3.3) for solving the DASLP model can also be readily utilized for solving the CLAP model. Indeed, the centralized model is an easier version to solve as the number of variables and constraints are far fewer when compared to the DASLP formulation, and furthermore, the CLAP formulation is a *cubic* multilinear program as compared to more complicated quartic DASLP model.

3.3.3 A Global Optimization Algorithm for DASLP

In order to make DASLP more amenable to algorithmic techniques, we reformulate the problem with the addition of several auxiliary variables. To begin with, let us denote,

$$y_i = z_i S_i \quad \forall \; i=1, \ldots \tag{3.9a}$$

$$g_{ik} = z_i q_k, \quad \forall (i,k). \tag{3.9b}$$

Next, to account for the trinomial terms $z_i S_i q_k$ and $z_i z_l S_l$ appearing in the model, define,

$$x_{ik} = (z_i S_i) q_k \equiv y_i q_k, \quad \forall (i,k). \tag{3.9c}$$

$$r_{il} = z_i (z_l S_l) \equiv z_i y_i, \quad \forall (i,l), i \neq l. \tag{3.9d}$$

Finally, to linearize the fourth-order terms in (3.6c), let,

$$w_{ilm} = z_i (z_l S_l) q_m \equiv z_i q_m, \quad \forall (i,l,m), i \neq l, \tag{3.9e}$$

$$v_{ikm} = (z_i S_i) q_k q_m \equiv x_{ik} q_m, \quad \forall (i,k,m). \tag{3.9f}$$

Now, using the substitutions described above yields the following equivalent *Reformulated DAS Location Problem* (**RDASLP**), given by:

RDSALP: Maximize $$\sum_{i=1}^{I} z_i \tag{3.10a}$$

Subject to $$\sum_{k=1}^{K} q_k = L, \tag{3.10b}$$

$$\begin{aligned} & K' T_{0i}^2 y_i + 2K' \sum_{k=1}^{K} T_{0i} T_{ki} x_{ik} + K' \sum_{k=1}^{K} \sum_{m=1}^{K} T_{ki} T_{mi} v_{ikm} - \sum_{\substack{l=1 \\ l \neq i}}^{I} T_{0i} T_{0l} r_{il} \\ & \qquad - \sum_{\substack{l=1 \\ l \neq i}}^{I} \sum_{m=1}^{K} T_{mi} T_{ml} w_{ilm} - z_i T_{0i} - \sum_{k=1}^{K} T_{ki} g_{ik} \geq 0 \end{aligned} \quad , \ \forall i \tag{3.10c}$$

$$\left. \begin{aligned} y_i &\leq S_i \\ y_i &\leq SNR_{\max} z_i \\ S_i - SNR_{\max} + SNR_{\max} z_i - y_i &\leq 0 \end{aligned} \right\} , \quad \forall\, i = 1, \ldots, I \tag{3.10d}$$

$$\left. \begin{aligned} g_{ik} &\leq z_i \\ g_{ik} &\leq q_k \\ -1 + z_i + q_k - g_{ik} &\leq 0 \end{aligned} \right\} , \quad \forall (i, k) \tag{3.10e}$$

$$\left. \begin{aligned} x_{ik} &\leq y_i \\ x_{ik} &\leq SNR_{\max} q_k \\ y_i - SNR_{\max} + SNR_{\max} q_k - x_{ik} &\leq 0 \end{aligned} \right\} , \quad \forall (i, k) \tag{3.10f}$$

$$\left. \begin{aligned} r_{il} &\leq SNR_{\max} z_i \\ r_{il} &\leq y_l \\ y_l - SNR_{\max} + SNR_{\max} z_i - r_{il} &\leq 0 \end{aligned} \right\} , \quad \forall (i, l), \ i \neq l, \tag{3.10g}$$

$$\left. \begin{aligned} w_{ilm} &\leq x_{lm} \\ w_{ilm} &\leq SNR_{\max} z_i \\ x_{lm} - SNR_{\max} + SNR_{\max} z_i - w_{ilm} &\leq 0 \end{aligned} \right\} , \forall (i, l, m), \ i \neq l \tag{3.10h}$$

$$\left. \begin{aligned} v_{ikm} &\leq x_{ik} \\ v_{ikm} &\leq SNR_{\max} q_m \\ x_{ik} - SNR_{\max} + SNR_{\max} q_m - v_{ikm} &\leq 0 \end{aligned} \right\} , \quad \forall (i, k, m) \tag{3.10i}$$

$$S_i \geq z_i, \qquad \forall\, i \tag{3.10j}$$

$$z_i,\ q_k,\ g_{ik} \in \{0, 1\}, \ \forall (i, k), \text{ and } 0 \leq y_i, x_{ik}, r_{il}, w_{ilm}, v_{ikm}, S_i \leq SNR_{\max}, \tag{3.10k}$$

where, the objective function (3.10a) maximizes the total number of admitted users, constraint (3.10b) restricts the number of DAS antenna locations to be a specified number L, constraint (3.10c) represents the linearized version of the SINR requirement under the substitutions (3.9a)-(3.9f), constraints (3.10d)-(3.10i) enforce the condition that at any *feasible* solution to RDASLP the auxiliary variables faithfully reproduce their corresponding nonlinear products (3.9a)-(3.9f), respectively, and finally constraints (3.10j) and (3.10k) define bounds for all the variables.

Observe that the original problem DASLP has of the order $\left(2I+K\right)$ variables and $2I$ constraints, whereas the reformulated problem RDASLP has $\left(3I+K\right)+\left(IK+I^2\right)+\left(I^2K+IK^2\right)$ variables and $3I+\left(2IK+I^2\right)+\left(I^2K+IK^2\right)$ constraints. Typically, as the number of users is much larger than the number of antennas, *i.e.*, $I >> K$, the number of variables and constraints in RDASLP can be approximated to be of order $O\left(KI^2\right)$. Thus, by the addition of a *polynomial* number of variables and constraints, we have obtained an equivalent *higher-dimensional mixed-integer 0-1 linear programming representation* of the original nonconvex binary multilinear program.

REMARK 4. The process adopted in our approach of linearizing Problem DASLP via the addition of new variables (given in (3.9a)-(3.9f)), and constraints (3.10d)-(3.10i) is based on the principle of *disjunctive programming*. We note here that there are several other schemes, such as quadrification, described in Shor (1990), which can be used to reduce Problem DASLP to a quadratic programming problem. This quadratic program can then be subsequently linearized using the process described in our approach. Another alternative would be to directly treat Problem DASLP as a *polynomial programming* problem, and use the methods described in Sherali and Tuncbilek (1992), to derive an equivalent LP representation. However, these schemes would lead to the addition of variables, of far greater orders of magnitude, which prohibits the use of these methods for large-scale instances.

For the sake of convenience, we shall denote the set of variables in RDASLP, with obvious vector notation, as

$$\xi \equiv \left(x, y, z, S, q, r, g, v, w\right) \tag{3.11}$$

Then, the following results are evident, where $v[\bullet]$ represents the optimal objective function value of the Problem$[\bullet]$.

LEMMA 1. *Let RDASLP-LP represent the linear programming (LP) relaxation of RDASLP. Then* $v[\text{RDASLP-LP}]$ *provides an upper bound on* $v[\text{RDASLP}]$ *and furthermore, if* $\bar{\xi} \equiv (\bar{x}, \bar{y}, \bar{z}, \bar{S}, \bar{q}, \bar{r}, \bar{g}, \bar{v}, \bar{w})$ *solves RDASLP-LP and satisfies the constraints (3.9a) – (3.9f), then it is optimal to DASLP*.

Proof. Obvious from construction of RDASLP. □

PROPOSITION 1. *Given a solution* $\bar{\xi} \equiv (\bar{x}, \bar{y}, \bar{z}, \bar{S}, \bar{q}, \bar{r}, \bar{g}, \bar{v}, \bar{w})$ *to RDASLP-LP, a feasible solution to DASLP can be obtained via the following heuristic.*

<u>Heuristic</u>

<u>*Step* 1:</u> Lexicographically arrange the vector $\bar{q}$ in nonincreasing order to obtain a vector, say $\hat{q} \equiv (\hat{q}_1, \ldots, \hat{q}_K, \hat{q}_{K+1}, \ldots, \hat{q}_L)$. Set $\hat{q}_1 = \hat{q}_2 = \cdots = \hat{q}_K = 1$, and set $\hat{q}_{K+1} = \cdots = \hat{q}_L = 0$. Go to Step 2. Set all z-variables as *free*.

<u>*Step* 2:</u> Resolve RDASLP with the q- and (some of the) z-variables fixed. (At the first iteration, none of the z-variables are fixed, as noted in Step 1.) If all the free z-variables take on binary values, then stop; the solution obtained is feasible to DASLP. Else, go to Step 3.

<u>*Step* 3:</u> Lexicographically arrange all the free z-variables in nondecreasing order, to obtain a vector, say $\breve{z}$. Select the lexicographically minimal 5% of the z-variables from $\breve{z}$, and fix them equal to zero. Assign the rest of the $\breve{z}$-variables as free, and go to Step 2.

Proposition 1 and Lemma 1 prompt a branch-and-bound algorithm for solving DASLP via RDASLP and its LP (continuous) relaxation, which we denote as RDASLP-LP. In this algorithm, we adopt a branch-and-bound process based on partitioning the search space by imposing the binary property on the 0-1 variables in the problem. Given any node sub-problem associated with the branch-and-bound tree, we solve the corresponding LP relaxation. Let $\bar{\xi} \equiv (\bar{x}, \bar{y}, \bar{z}, \bar{P}, \bar{q}, \bar{r}, \bar{g}, \bar{v}, \bar{w})$ be the optimal solution obtained for this node subproblem. If the condition of Lemma 1 holds true, then we will also have solved the node subproblem of RDASLP. Otherwise, we could apply Proposition 1 to (possibly) update the incumbent solution, and branch at this node by partitioning the search space as follows.

Branching Rule. Find the discrepancies: $\theta_{\hat{i}} = \max\limits_{i=1,\ldots,I} \left\{ \left| \bar{y}_i - \bar{z}_i \bar{S}_i \right| \right\}$, $\lambda_{\hat{i}\hat{k}} = \max\limits_{(i,k)} \left\{ \left| \bar{g}_{ik} - \bar{z}_i \bar{q}_k \right| \right\}$, and let, $\mu \equiv \max\left\{ \theta_{\hat{i}}, \lambda_{\hat{i}\hat{k}} \right\}$. If $\mu = \theta_{\hat{i}}$, then for the corresponding variable that achieves this maximum value, impose the dichotomy $z_{\hat{i}} = 0$ and $z_{\hat{i}} = 1$ to generate the children nodes; otherwise, if $\mu = \lambda_{\hat{i}\hat{k}}$, then for the corresponding variable that achieves this maximum value, impose the dichotomy $q_{\hat{k}} = 0$ and $q_{\hat{k}} = 1$, breaking ties in favor of the variable with the least index.

At each stage t of this branch-and-bound procedure, $t = 0, 1, \ldots, T$, we will have a set of non-fathomed or *active nodes* A_t, where each node $a \in A_t$ is indexed by a set of *fixed variables*. (To initialize, at $t = 0$, the set $A_0 = \{0\}$, with none of the variables being fixed.) For each node $a \in A_t$, an upper bound UB_a will be given by $\nu\left[\text{RDASLP-LP}^a\right]$, where, in our notation, $\nu\left[\text{RDASLP-LP}^0\right] \equiv \nu\left[\text{RDASLP-LP}\right]$. As a result, the (*global*) *upper bound* at stage t for RDASLP (and equivalently, DASLP) is given by

$$\text{UB} \equiv \underset{a \in A_t}{\text{maximum}} \left\{ \text{UB}_a \right\}. \tag{3.12}$$

Whenever any upper bounding node subproblem is solved, we can apply Proposition 1 in order to derive a lower bound (and possibly update the incumbent solution) on the overall problem DASLP, where the corresponding lower bound value is given by (3.10a). Accordingly, let $\left(z^*, S^*,\ q^*\right)$ be the best such incumbent solution found, having an objective value of ν^*. Naturally, whenever $UB_a \leq \nu^*$, we can fathom node *a*. (Practically, we can fathom node *a* whenever $UB_a \leq \nu^*\left(1+\varepsilon\right)$, for some optimality tolerance $\varepsilon > 0$.) Hence, the active nodes at any stage *t* would satisfy $UB_a > \nu^*,\ \ \forall\, a \in A_t$. From this set of active nodes, we now select a node $a\left(t\right)$ that yields the greatest upper bound, *i.e*., for which $UB_{a(t)} \equiv$ UB as given by (3.12). Note that for the corresponding solution

$$\xi^{a(t)} \equiv \left(x^{a(t)},\ y^{a(t)},\ z^{a(t)},\ S^{a(t)},\ q^{a(t)},\ r^{a(t)},\ g^{a(t)}, v^{a(t)}, w^{a(t)}\right) \tag{3.13}$$

to RDASLP-LP$^{a(t)}$, we could not possibly have $\bar{y}_i = \bar{z}_i\bar{S}_i,\ \ \forall\, i$, and $\bar{g}_{ik} = \bar{z}_i\bar{q}_k,\ \ \forall\left(i, k\right)$, because then, from Lemma 1, $\xi^{a(t)}$ would be feasible to RDASLP$^{a(t)}$, thereby yielding $UB_{a(t)} \leq \nu^*$, which is a contradiction. Therefore, we can find a branching variable and partition the node subproblem based on the proposed branching rule, and proceed to the next stage.

REMARK 5. From a continuous relaxation point of view, RDASLP can be further tightened by the following approach. Multiply the constraint (3.10b) by the terms z_i and $\left(1-z_i\right)$, $\forall\, i = 1,\ldots, I$, and furthermore, multiply (3.10j) by z_i. (Note that, multiplying (3.10j) by $\left(1-z_i\right)$ would merely yield the constraint $y_i \leq S_i,\ \ \forall\, i$, which already exists in (3.10d).) Under the substitution (3.9b), this yields the constraints:

$$\sum_{k=1}^{K} g_{ik} - Lz_i = 0, \quad \forall\left(i, k\right) \tag{3.14a}$$

$$\sum_{k=1}^{K} q_k - \sum_{k=1}^{K} g_{ik} - L + Lz_i = 0, \quad \forall (i, k) \tag{3.14b}$$

$$y_i \geq z_i, \quad \forall i = 1, \ldots, I \ . \tag{3.14c}$$

Denote the formulation with objective function (3.10a), and constraints (3.14a)-(3.14c), (3.10c)-(3.10i), and (3.10k), as **RDASLP-2**. This new formulation contains an additional $O(KI)$ number of constraints as compared to RDASLP, but provides a tighter formulation because any feasible solution to the LP relaxation of RDASLP-2 is also feasible to the LP relaxation of RDASLP, but not vice versa. In our computations, we explore the relative merits of this alternate formulation, and provide extensive results that test the promise of all the proposed models. Note that a branch-and-bound algorithm identical to the one described above can be used to solve both formulations.

3.3.4 An Approximate System Model

In lieu of the extremely large number of variables and constraints that can potentially be generated in the formulation of RDASLP, we now consider an *approximate approach* that serves to alleviate some of the computational difficulties faced by the branch-and-bound algorithm by significantly reducing the size of the model formulation. Here, we begin with the basic assumption that the cell coverage in the system is small enough such that the maximum transit power constraint practically becomes inactive. Then, we make two further simplifications to the formulation presented in Section 3.3.1 to develop an approximate, albeit faster, methodology for locating distributed antenna within this single cell system.

Consider the denominator in (3.3), where, recognizing that $|\omega_{0i}|^2 + \sum_{k=1}^{K} |\omega_{ki}|^2 \equiv 1, \forall\, i$, we get:

$$D_i \equiv |\omega_{0i}|^2 \sum_{\substack{l=1 \\ l \neq i}}^{I} z_l P_l T_{0l} + \sum_{k=1}^{K} q_k |\omega_{ki}|^2 \sum_{\substack{l=1 \\ l \neq i}}^{I} z_l P_l T_{kl} + \eta W \ . \tag{3.15}$$

In the first simplification, suppose that in (3.15), $|\omega_{0i}|^2$ and $|\omega_{ki}|^2$ are the same for all users. Denoting these weights by $|\omega_0|^2$ and $|\omega_k|^2$, respectively, we assume that these common values can be computed as the weighted average of $|\omega_{0i}|^2$ and $|\omega_{ki}|^2$ over all admitted users, *i.e.*,

$$|\omega_0|^2 \equiv \frac{1}{\sum_{i=1}^{I} z_i}\left[\sum_{i=1}^{I}|\omega_{0i}|^2 z_i\right] \text{ and } |\omega_k|^2 \equiv \frac{1}{\sum_{i=1}^{I} z_i}\left[\sum_{i=1}^{I}|\omega_{ki}|^2 z_i\right] . \quad (3.16)$$

Using these constraints weights (3.16), the denominator in (3.3) can now be expressed as:

$$D_i \equiv |\omega_0|^2 \sum_{\substack{l=1 \\ l\neq i}}^{I} z_l P_l T_{0l} + \sum_{k=1}^{K} q_k |\omega_k|^2 \sum_{\substack{l=1 \\ l\neq i}}^{I} z_l P_l T_{kl} + \eta W$$

$$= |\omega_0|^2 \sum_{l=1}^{I} z_l P_l T_{0l} + \sum_{k=1}^{K} q_k |\omega_k|^2 \sum_{l=1}^{I} z_l P_l T_{kl} - z_i P_i T_{0i} |\omega_0|^2 - \sum_{k=1}^{K} q_k z_i P_i T_{ki} |\omega_k|^2 + \eta W \quad (3.17)$$

Note that (3.17) is independent of the index i except the third and fourth terms. In the second part of our approximation, we assume that these terms are small compared to the first and second terms for a large system capacity. Thus, ignoring the third and fourth terms in (3.17), we get a common denominator D, given by,

$$D \equiv |\omega_0|^2 \sum_{l=1}^{I} z_l P_l T_{0l} + \sum_{k=1}^{K} q_k |\omega_k|^2 \sum_{l=1}^{I} z_l P_l T_{kl} + \eta W \quad (3.18)$$

Using this common denominator, we can rewrite (3.3) as

$$\mathrm{SINR}_i = \frac{z_i \dfrac{W}{R} P_i \left(T_{0i} + \sum_{k=1}^{K} q_k T_{ki}\right)}{D}, \quad (3.19)$$

Which in turn implies that the transmit powers must meet the following *minimum requirement*, given by:

$$P_i \geq \frac{R\Gamma}{W} D \left(\frac{1}{T_{0i} + \sum_{k=1}^{K} q_k T_{ki}}\right) z_i . \quad (3.20)$$

Substituting this minimum power level, as derived in (3.20), for all admitted users i into (3.18), we get the following *implicitly* defined equation for D:

$$D = \left[|\omega_0|^2 \sum_{l=1}^{I} \left(\frac{z_l T_{0l}}{T_{0l} + \sum_{k=1}^{K} q_k T_{kl}} \right) + \sum_{k=1}^{K} q_k |\omega_k|^2 \sum_{l=1}^{I} \left(\frac{z_l T_{kl}}{T_{0l} + \sum_{k=1}^{K} q_k T_{kl}} \right) \right] \frac{D}{K'} + \eta W \, , \quad (3.21)$$

where, as defined above, $K' = W/R\Gamma$. Solving (3.21) for D, and subsequently substituting the value of $|\omega_0|^2$ and $|\omega_k|^2$ (given by (3.16)), we can show that D (and consequently, $P_i, \quad \forall i = 1,...,I$) is nonnegative if and only if,

$$K' \sum_{i=1}^{I} z_i - \left(\sum_{l=1}^{I} \frac{T_{0l} z_l}{T_{0l} + \sum_{k=1}^{K} q_k T_{kl}} \right)^2 - \sum_{k=1}^{K} q_k \left(\sum_{l=1}^{I} \frac{T_{0l} z_l}{T_{0l} + \sum_{m=1}^{K} q_m T_{ml}} \right)^2 \geq 0. \quad (3.22)$$

We are now ready to formulate the *Approximate Distributed Antenna Location Problem* (**ADASLP**), where the new QoS constraint for locating distributed antennas is now enforced by (3.22). Note that this feasibility constraint is *common* for all demand nodes.

ADSALP: Maximize $\sum_{i=1}^{I} z_i$ (3.23a)

Subject to $\sum_{k=1}^{K} q_k = L,$ (3.23b)

$$K' \sum_{i=1}^{I} z_i - \left(\sum_{l=1}^{I} \frac{T_{0l} z_l}{T_{0l} + \sum_{k=1}^{K} q_k T_{kl}} \right)^2 - \sum_{k=1}^{K} q_k \left(\sum_{l=1}^{I} \frac{T_{0l} z_l}{T_{0l} + \sum_{m=1}^{K} q_m T_{ml}} \right)^2 \geq 0, \quad (3.23c)$$

$$z_i, q_k \in \{0,1\}, \quad \forall (i,k). \quad (3.23d)$$

Here, the objective function (3.23a) seeks to maximize the total number of admitted users, constraint (3.23b) restricts the number of DAS antenna locations to be a specified number *L*, constraint (3.23c) imposes the new QoS requirement that $D \geq 0$ for all admitted users, and finally, constraint (3.23d) specifies the binary nature of the (z, q)-variables. Note that, while RDASLP described earlier was formulated over the variables (z, S, q), this approximate model, on the other hand, has reduced the size of the formulation to the order of $(I + K)$ binary variables and only two constraints, by virtue of eliminating the continuous SNR-variables, *S*. More notably, the complicating constraint (3.23c) is nonconvex, thereby rendering ADASLP as a cubic nonconvex program.

REMARK 6. In the spirit of the exact approach, we can indeed design a (globally) convergent branch-and-bound algorithm for the approximate formulation (3.23a)-(3.23d) as well, through the addition of suitably defined auxiliary variables and constraints. However, in this linearization process, we would need to contented with the squares of summation terms that exist in constraint (3.23c), which in turn would lead to a branch-and-bound algorithm that requires partitioning the search space defined in terms of several *continuous* variables (Sherali and Tuncbilek, 1992). While such algorithmic schemes can be customized for the approximate formulation, the strength of these methods is highly dependent on accurately computing the lower and upper bounds of the auxiliary variables, which can prove to be difficult in our case. Hence, we defer solving ADASLP to one of the many commercial nonlinear programming solvers that are readily available, *e.g.*, BARON (Sahinidis, 1996) or OQNLP (Ugray *et al.*, 2005). In our computations, we also test the strength of this approximate approach provide insights, and compare these solutions to ones obtains from the exact approach.

3.4 Computational Results

In our computations, we defined the region $\Re$ to be a symmetrical square of area one-square kilometer centered at the origin, *i.e.*, $\Re \equiv \{(x, y) : (x, y) \in [-0.5, 0.5] \times [-0.5, 0.5]\}$. In this region, the base station (BS) is located at the origin, and four points were randomly chosen via a uniform distribution as the set of possible candidate DAS antenna locations. Now, given user *i*'s Cartesian coordinates $(\alpha_i, \beta_i) \in \Re$, let $d_{0i} \equiv \sqrt{\alpha_i^2 + \beta_i^2}$ denote the distance from the base station to demand node *i*. Then, we compute the path gain from user node *i* to the base station as:

$$T_{0i} = \begin{cases} H_{BS}\left(\dfrac{d_{0i}}{d_{ref}}\right)^{-2} 10^{\chi_1/10}, & d_{0i} \le b_{BS} \\ H_{BS}\left(\dfrac{d_{0i}}{d_{ref}}\right)^{-4} 10^{\chi_1/10}, & d_{0i} > b_{BS} \end{cases}, \tag{3.24}$$

where, $H_{BS} = 10$, $b_{BS} = 100\text{m}$, $d_{ref} = 1\,\text{km}$, and χ_1 is a Gaussian random variable with parameters $\left(\mu_1, \sigma_1^2\right) = \left(0,8\right)$. Similarly, if d_{ki} represents the distance from demand node i to candidate location k, then the path gain from user i to a DAS antenna at candidate location k is given by,

$$T_{ki} = \begin{cases} H_{DAS}\left(\dfrac{d_{ki}}{d_{ref}}\right)^{-2} 10^{\chi_2/10}, & d_{ki} \le b_{DA} \\ H_{DAS}\left(\dfrac{d_{ki}}{d_{ref}}\right)^{-4} 10^{\chi_2/10}, & d_{ki} > b_{DA} \end{cases}, \tag{3.25}$$

where, $H_{DA} = 10$, $b_{DA} = 100\,\text{m}$, $d_{ref} = 1\,\text{km}$, and χ_2 is also a Gaussian random variable with parameters $\left(\mu_2, \sigma_2^2\right) = \left(0,4\right)$. Furthermore, we set $SNR_{\max} = 10$.

Although we set the path gains as specified above, the models studied here can also accommodate any other type of path gain model, *e.g.*, those that account for antenna directivity, spatial correlation between demand node and a particular DAS antenna, *etc*. Using the above defined data generation parameters, we test the proposed models and algorithms for five different demand scenarios. These demand scenarios are generated by varying the number of users in the region, the number of “hotspots” (HS), the length of the hotspot, and the percentage of users selected from the hotspot. The different scenarios are as summarized below:

Scenario 1: 75 users, uniformly distributed across $\Re$.

Scenario 2: 100 users, uniformly distributed across $\Re$.

Scenario 3: 75 users, 1 HS at $\left(0.1, 0.35\right)$, with length 0.25, 50% uniform, 50% from HS.

Scenario 4: 100 users, 2 HS at $(0.1, 0.2)$ and $(-0.1, -0.2)$, with length 0.2, 50% uniform, 25% from each HS.

Scenario 5: 100 users, 2 HS at $(0.1, 0.2)$ and $(-0.1, -0.2)$, with length 0.2, 75% uniform, 12.5% from each HS.

All of these computations were performed using the GAMS modeling language (version 23.9) on an Intel Core i7 2.80 GHz processor with 6GB RAM in a Windows environment. To begin with, we solved the 0-1 mixed-integer polynomial program, namely DASLP, using the commercial global optimization solvers BARON [21] and OQNLP [30], and the best resulting solution was recorded. Then, the linearized models, RDASLP and RDASLP-2, were solved using the proposed branch-and-bound algorithm, in conjunction with the commercial software CPLEX (version 12.0), which was used to solve the underlying linear programs in the enumeration tree. Finally, the approximate model ADASLP was also solved using BARON and OQNLP. A resource limit of 50000 seconds was utilized for each formulation, and in the event that the optimal solution was not obtained in that time limit, the best known (feasible) solution is provided in the results below. We set the number of antennas to be deployed as two out of a possible set of four candidate locations, *i.e.*, $L = 2$, and $K = 4$.

Table 17 provides the computational results for Scenario 1, where the "*q*"-column indicates the antenna locations selected by a particular formulation, and v denotes *best* (not necessarily the optimal) objective function value attained, *i.e.*, the number of users admitted into the system. In Scenario 1, Problem DASLP has 79 discrete variables and 76 continuous variables, and the best objective function value obtained was 11, with corresponding antenna locations 1 and 2. Solving the approximate model, ADASLP (with only 79 discrete variables), leads to an objective function value of 32, with antennas 2 and 3 being deployed. (In ADASLP, this solution is found in 4830 seconds, which is less than the prescribed resource limit.)

Table 17: Computational results for Scenario 1

Problem	*v*	*q*	# of variables	Solver	Iteration # where *v* is found	CPU time (s)	First integer solution /Iteration #	Total Iteration #
DASLP	11	1,2	155 79 discrete	BARON	156	12640	4/30	1,760
RDASLP	44	2,3	29780 79 discrete	CPLEX	84793	20640	36/21261	153,632
RDASLP-2	44	1,2	29780 79 discrete	CPLEX	53725	11674	43/29000	161,659
ADASLP	32	2,3	80 79 discrete	OQNLP	41843	4830	6/19386	88,243

We also recorded the iteration where the *first* incumbent (integer feasible) solution was obtained, and for Problem DASLP, the best solution was obtained at node 156, and there was no improvement thereafter. The first incumbent solution found in ADASLP had an objective function value of 6, which was found at iteration 19386. In another statistic, the iteration number where the best solution was found is reported, and for Problems DASLP and ADASLP, this occurs at iterations 156 and 41843, respectively.

Next, to determine (globally) optimal solutions and to test the efficacy of the linearized formulations, we solved Problems RDASLP and RDASLP-2 via the proposed branch-and-bound algorithm. The number of variables is significantly larger in the linearized models (as compared to the original nonlinear formulation), with 29780 total variables, out of which 79 are discrete. The optimal solution obtained (using both linearized formulations) is 44 (out of 75 users), with both models converging to the optimum within the prescribed time limit. Problem RDASLP performed relatively better in Scenario 1 in terms of the number of nodes enumerated, but conversely, the iteration numbers at which first incumbent solution and v are found are correspondingly smaller in RDASLP-2. Using the RDASLP-2 formulation, the branch-and-bound algorithm determined the first integer solution (given by 43) at iteration 29,000 and v (given by 44) at iteration 53,725. On the other hand, the total CPU time required by RDASLP-2 is nearly 44% lesser than the CPU time consumed by

RDASLP, which indicates that due to the tightness of the LP relaxations in Problem RDASLP-2, the computational effort required per node is significantly lesser as opposed to Problem RDASLP.

Figure 7 displays the scatter plot for Scenario 1, indicating the activated antennas for each of the model formulations, where the candidate antenna locations are indexed as shown in the figure. For example, note that antenna 2 is deployed by all the models. At optimum, models RDASLP and RDASLP-2 admit 58.7% of the users, whereas Problems ADASLP and DASLP admit only 42.7% and 14.6%, respectively. Hence, based on Scenario 1, it is clearly seen that the linearized models, in conjunction with the tailored algorithm, performed significantly better than merely solving the nonlinear models using a commercial solver. Moreover, Problem ADASLP was relatively successful as compared to Problem DASLP, in terms of the number of admitted users as well as the consumed CPU time, which was a notable result.

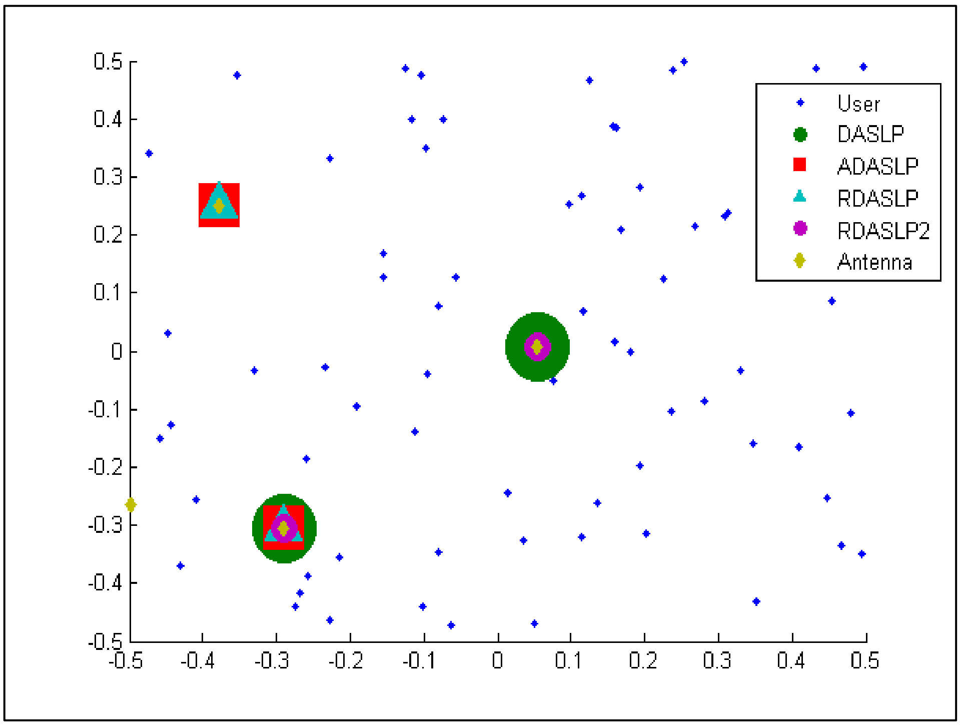


Figure 7: Scatter plot for Scenario 1

Finally, to gauge the relative increase in capacity by employing a DAS framework as compared to its centralized counterpart, we solved the linearized version of Problem CLAP, obtained by employing an identical variable substitution and algorithmic strategy seen in Section 3.3. The corresponding optimal objective function value of Problem CLAP was recorded as 26, *i.e.*, co-locating antennas at the base station admits only 34.66% of users simultaneously as compared to the 58.7% of users admitted by employing a distributed antenna system network. However, owing to the relative smaller formulation size, solving Problem CLAP only consumes 67% of CPU time, when compared to Problem RDASLP-2. Having established the efficacy of the proposed DAS framework in achieving a higher system capacity, we now test the merits of this approach on different user network configurations as listed in our scenario descriptions.

Recognizing the similarity of Scenarios 1 and 2 (generated by using a uniform user distribution throughout the region), we next solved Scenario 2 to reinforce the conclusions drawn from Scenario 1. As seen from the statistics recorded in Table 18, these computational results are consistent with our previous findings in Scenario 1, with Problems RDASLP and RDASLP-2 obtaining the globally optimal objective function values. The only change we noticed was that, in Scenario 2, the objective function value of the first incumbent solution obtained (= 43) when solving Problem RDASLP was greater than the obtained (= 41) by solving Problem RDASLP-2. As in Scenario 1, the optimal objective function value obtained from Problem ADASLP is suboptimal, but nevertheless, it continues to dominate Problem DASLP. (In this case, Problem DASLP failed to converge and did not produce any feasible solution.) The user configurations and optimal antenna deployments corresponding to the different formulations are displayed in Figure 8.

Table 18: Computational results for Scenario 2

Problem	*v*	*q*	# of variables	Solver	Iteration # where *v* is found	CPU time (s)	First integer solution /Iteration #	Total Iteration #
DASLP	-	-	205 104 discrete	OQNLP	-	50000	-	8,611
RDASLP	43	1,3	52,205 104 discrete	CPLEX	93974	32480	43/59944	210,057
RDASLP-2	43	2,3	52,205 104 discrete	CPLEX	87462	22480	41/53826	193,519
ADASLP	33	2,3	105 104 discrete	BARON	37	6300	32/2800	133,543

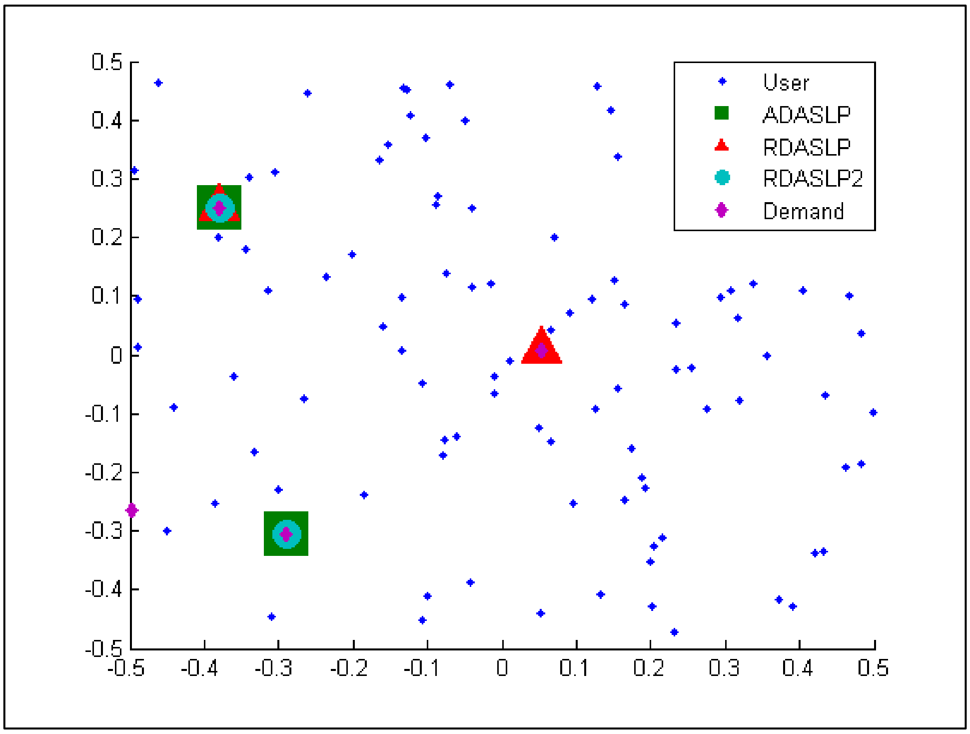


Figure 8: Scatter plot for Scenario 2

The computational results showed some different trends for Scenarios 3 through 5, which exhibit concentrated user demand at the hotspots, and the rest of the demand is uniformly spread out through the region. In Scenario 3, Problem DASLP outperformed the approximate model ADASLP by admitting 19 users (corresponding to antennas 3 and 4), whereas Problem ADASLP failed to find any solution due to numerical errors. As Problem

ADASLP is a fractional program, the denominator terms tend to be numerically close to zero at certain points in the feasible region, leading to convergence errors using a nonlinear solver. Moreover, Problem DASLP was solved using OQNLP and not BARON, as reported for the prior cases. Once again, the linearized models dominated the original nonconvex model, by determining the global optimal objective function value of 49. However, it should be noted that Problem RDASLP-2 found the first integer solution (= 42) in fewer iterations as compared to the first integer solution determined by Problem RDASLP (= 38). Moreover, keeping in sync with the user distribution (refer Figure 9), antenna 1 is not deployed in any of the models, and furthermore, 65.3% of the users were admitted into the system at optimality, whereas Problem DASLP admitted only 25.3% of the users in this scenario.

Table 19: Computational results for Scenario 3

Problem	***v***	***q***	**# of variables**	**Solver**	**Iteration # where *v* is found**	**CPU time (s)**	**First integer solution /Iteration #**	**Total Iteration #**
DASLP	19	3,4	155 79 discrete	OQNLP	1,159	3426	19/1159	3,615
RDASLP	49	2,3	29,780 79 discrete	CPLEX	167,406	44480	38/54442	406,479
RDASLP-2	49	2,3	29,780 79 discrete	CPLEX	209,649	32480	42/93736	574,995
ADASLP	-	-	-	-	-	-	-	-

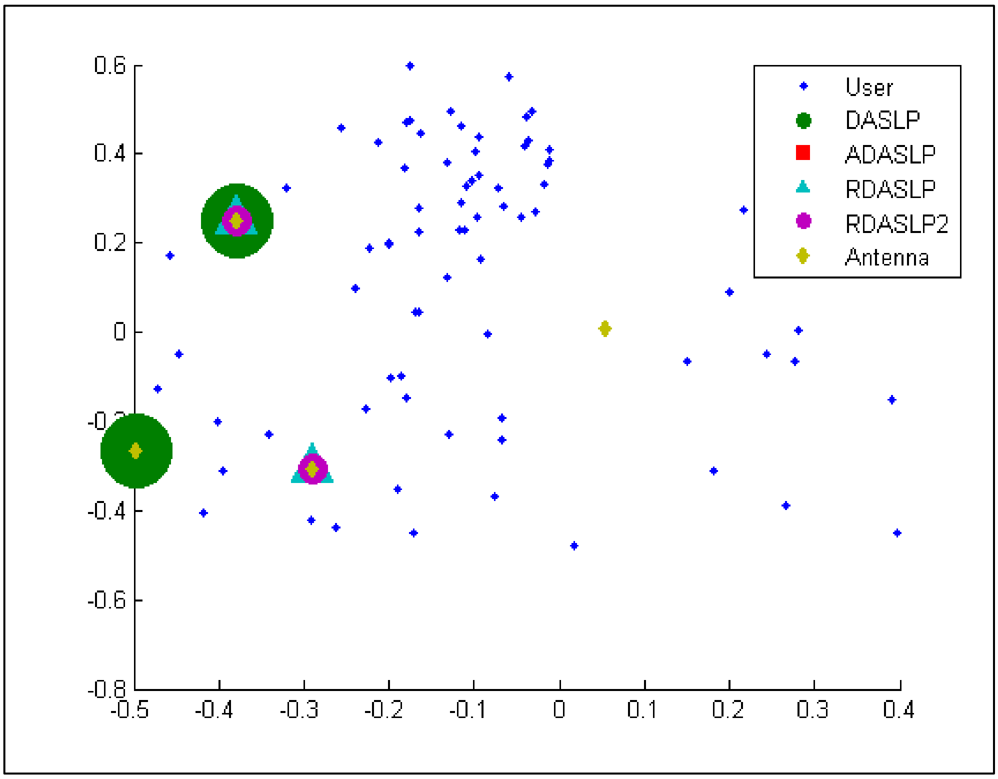


Figure 9: Scatter plot for Scenario 3

In Scenario 4, the tightness of LP relaxation for Problem RDASLP-2 leads to it being the only formulation to achieve the global optimal solution, given by 44 (out of total of 100 users). This is also the first occasion where we observed a difference in the objective function value between RDASLP (39) and RDASLP-2. However, they deployed exactly the same antennas, namely antenna 1 and 3. The approximate model ADASLP also deployed the optimal set of antennas in Scenario 4, while admitting only 31 users. On the other hand, Problem DASLP deployed antenna 3 and 4, and the best objective function value of 2 was found during processing, and no better incumbent solution was obtained before reaching the resource limit of 50,000 seconds. These computational results are summarized in Table 20 and Figure 10.

Table 20: Computational results for Scenario 4

Problem	*v*	*q*	# of variables	Solver	Iteration # where *v* is found	CPU time (s)	First integer solution /Iteration #	Total Iteration #
DASLP	2	3,4	205 104 discrete	BARON	-	50000	-	1
RDASLP	39	1,3	52,205 104 discrete	CPLEX	211,864	42400	39/21864	247,305
RDASLP-2	44	1,3	52,205 104 discrete	CPLEX	189,145	38780	36/53545	210,026
ADASLP	31	1,3	105 104 discrete	BARON	476	50000	27/46	236,246

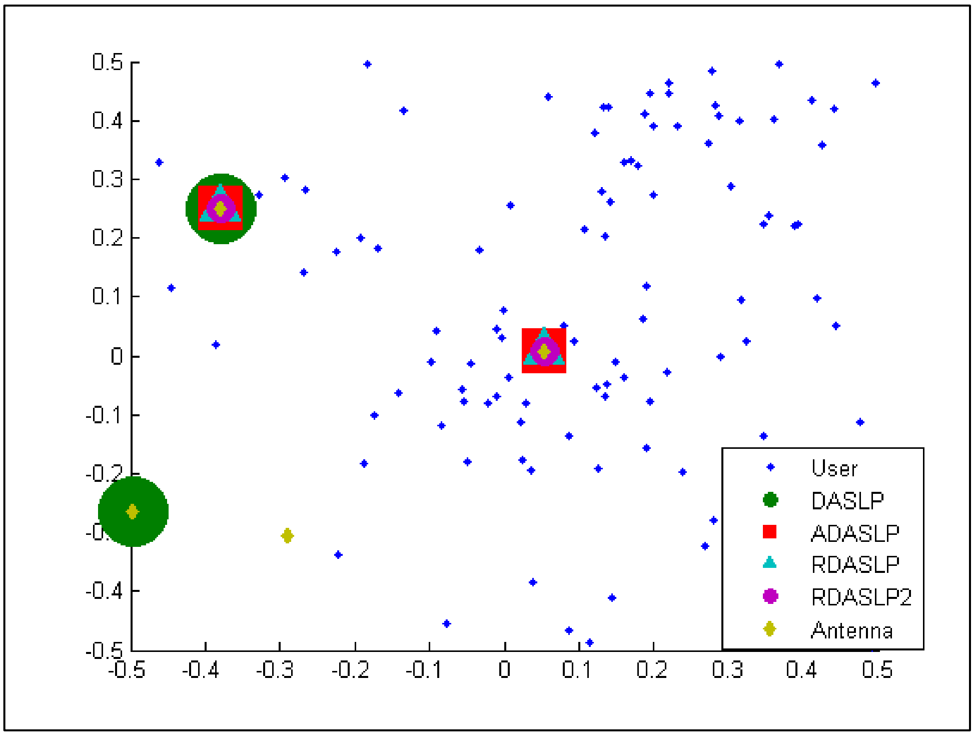


Figure 10: Scatter plot for Scenario 4

We see similar results for Scenario 5 as compared to Scenario 4. Problem RDASLP-2 attained the maximum objective function value, given by 43 (out of 100 users). In this Scenario, all of the other models determined the optimal set of antennas, except for Problem DASLP, which failed to determine any integer solution. Computational results for Scenario 5 are displayed in Table 21, and Figure 11 displays the corresponding user distribution and optimal antenna deplotments for each model.

Table 21: Computational results for Scenario 5

Problem	*v*	*q*	# of variables	Solver	Iteration # where *v* is found	CPU time (s)	First integer solution /Iteration #	Total Iteration #
DASLP	0	-	205 104 discrete	OQNLP	-	50000	-	5,426
RDASLP	41	1,3	52,205 104 discrete	CPLEX	198,760	46,343	27/107588	228,476
RDASLP-2	43	1,3	52,205 104 discrete	CPLEX	189,905	38,298	39/22841	208,254
ADASLP	29	1,3	105 104 discrete	BARON	163	50000	26/136	284,324

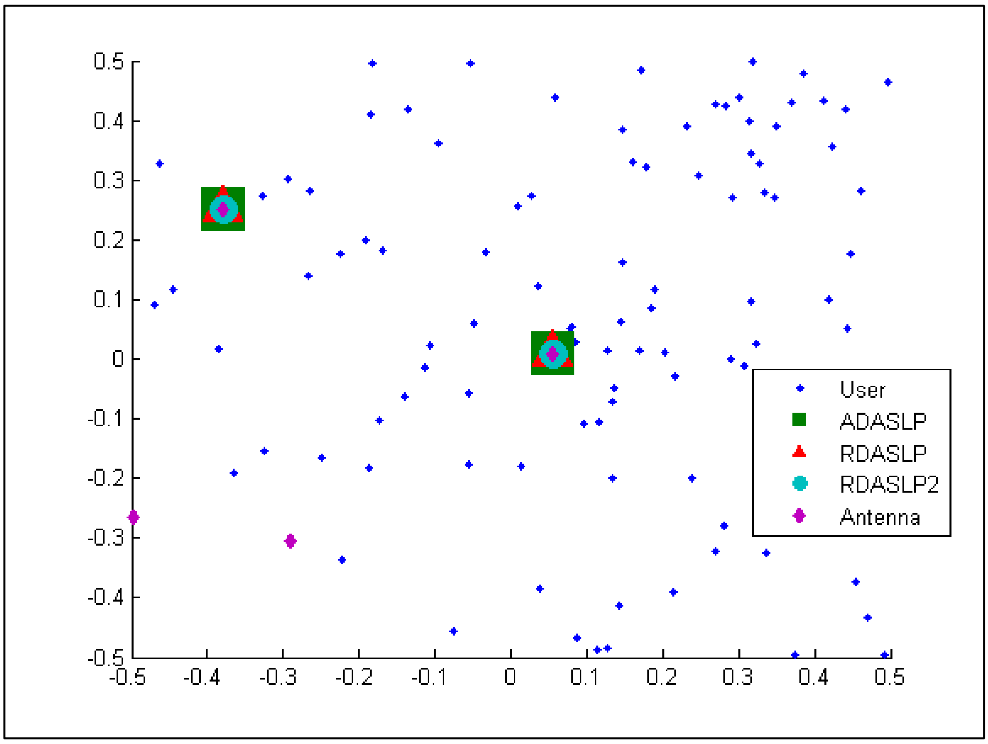


Figure 11: Scatter plot for Scenario 5

Summarizing the results from all of the five scenarios, the linearized models in conjunction with the proposed algorithm clearly outperformed the original nonlinear model, Problem DASLP. Indeed, the approximate system model, ADASLP, was also relatively better at obtaining good solutions, with significantly lesser computational effort, when compared to

Problem DASLP, even in cases when DASLP failed to find a solution. This validates ADASLP being a viable alternative, particularly for large-scale instances wherein DASLP is encumbered due to the complexity of the problem size. In comparing the linearized models, the computational results were similar, but Problem RDASLP-2 did yield better results in some cases, which we attribute to the additional (tightened) constraints incorporated into the formulation.

3.5 Extensions and Conclusions

In this chapter, we posed and solved the problem of determining a set of distributed DAS antennas for maximizing the total number of users (user capacity) in a CDMA system such that each admitted user satisfies the minimum SINR requirement while respecting the maximum transit power constraint. This complex problem was formulated as a mixed-integer 0-1 multilinnear program, where the largest nonlinear (and nonconvex) term is a quartic-degree polynomial. We reformulated this nonconvex problem using auxiliary variables and constraints to yield and equivalent higher dimensional mixed-integer 0-1 linear programming problem. A tailored branch-and-bound algorithm, in conjunction with an intelligent heuristic, was designed to solve the linearized problem. The computational results for different user distribution scenarios demonstrated the efficiency of the proposed algorithm, and the global optimum was determined for all the test instances, even in cases where well-known commercial global optimization solvers converged to poor suboptimal solutions.

From an algorithmic and computational point of view, a rigorous analytical framework, such as the one developed in this paper, is very useful for service providers opting for a DAS-based network solution. The exact global optimization framework, accompanied by a specialized linearization strategy and search heuristic techniques, presented in our work leads to tight lower and upper bounds at every iteration, thereby leading to the desired algorithmic convergence. Moreover, a generic approach such as the one prescribed in this paper can form the basis for addressing a host of problems arising in cellular network location problems.

Our modeling approach is valid for both single-cell and multi-cell systems, in which out-of-cell interference can be represented as Gaussian noise. Our results hold for systems where single-rate users are present and either non-fading or "average" fading phenomena can be used to adequately model system dynamics for a deployment solution. When a more rigorous treatment of inter-cell interference is required, both the optimization models presented here can be extended to determine optimal distributed antenna locations in larger multi-cell systems, although we believe the exact approach will become increasingly complex. The approximate model, however, will have reasonable complexity and thus represents a viable, if not desirable approach, in designing DASs for multi-cell CDMA systems.

A computationally reasonable alternative to modeling the multi-cell scenario uses the coupling equation system and the interference coupling matrix which can be constructed for each user configuration, *e.g.*, see Geerdes (2008) and Nawrocki et al. (2006). A potential avenue for future research would be to study the impact of incorporating this interference within the modeling framework for the DAS location problem (as developed in this paper) for a multi-cell scenario. As distributed antenna systems offer the opportunity to reduce distances between transmitters (demand nodes) and receivers (DAS and base station antennas), we expect that the resulting reduction in transmit power levels will translate to lesser total interference from one cell to adjacent or neighboring cells. In fact, from a wider perspective, a multi-cell DAS system very much resembles a traditional CDMA system with single-antenna base stations (of varying heights), where each user engages in soft-handoff with L neighboring base stations. Thus, we expect that a more detailed treatment of a multi-cell DAS system to deliver similar benefits as soft-handoff seen in traditional CDMA networks.

Another important avenue for future research stems from solving the DAS problem with not only out-of-cell interference but in the presence of in-cell *femtocells*. Closed femtocell deployments, in which users (and not service providers) install small, local base stations, can be considered as in-cell interferers whose impact on system capacity can be mitigated by deploying DAS systems. The interference model in such scenarios will be different than the work presented in this paper (and even from those in a more rigorous multi-cell DAS problem), and is the focus of our ongoing research.

Chapter 4

A Discrete Optimization Framework for DAS Locations in CDMA Cellular Networks with Femtocells

Due to the rapid development of wireless technology, enabling not only voice but also data and multimedia options, cellular phones have become an indispensable part of our everyday life. But an often recurring complaint among consumers concerns the imperfect coverage of current cellular networks, and this problem is particularly acute in indoor locations leading to an increased probability of signal loss (Chandrasekhar *et al.*, 2008). In such situations, a new and upcoming cellular network technology known as "*femtocells*" provides a promising option for cellular companies to improve their coverage, and provide high quality services in a cost effective manner.

However, as with any newly developed technology, femtocells face numerous issues related to widespread adaptability that have yet to be resolved. Such problems include interference management, handoff mechanisms, security concerns, *etc.* (Ho and Claussen, 2007). For example, when an unauthorized cell phone is within the coverage of a femtocell, it would not be able to connect to the femtocell and instead would rely on obtaining coverage directly from the distant macrocell base station. This user's mobile connectivity would experience significant negative interference from the femtocell (Claussen, 2007). Thus further research is required to understand the steps that need to be taken to alleviate interference from neighboring femtocells, and consequently improve indoor coverage.

In Chapter 3, we have already introduced some of the key elements of cellular networks relevant to this research, including a basic description of the workings of a cellular network, distributed antenna systems, interference characteristics in CDMA networks, handoff mechanisms. Next, we introduce another crucial element, namely "femtocells" in the following discussion. The remainder of this chapter is organized as follows. In Section 4.1, we first present the definition of "femtocell", along with its advantages and disadvantages. In

Section 4.2, we present a detailed description of the underlying mathematical model and derive the exact formulation for solving the problem under consideration. In this context, we also scour through some of the literature that closely resembles our current work. Then in Section 4.3, we describe the set-up for our computational experiments, implement the formulation in commercially available optimization software, and analyze the results obtained. Some computational experiences are presented in Section 4.4. Finally, in Section 4.5, some concluding remarks and directions for future research are mentioned.

4.1 Introduction

A "*femtocell*" is a small, simple, "plug-and-play" cellular base station. Depending on the transmit power and surrounding interferences, a femtocell usually has a range of about 10 meters (according to the Femtocell Forums). By comparison, microcells, which are used by service providers to minimize the gaps in coverage between macrocell base stations, usually have a range of two kilometers.

A femtocell is usually designed to be used either at home or within a small enterprise setting. This is because current femtocells are only capable of supporting up to sixteen active mobile users concurrently. A femtocell connects to the network provider directly (available from the Femtocell Forums), bypassing the antenna elements of a CDMA network. This can be done either through broadband, or through the use of an external antenna such as a satellite dish. It allows service providers to extend their service coverage to areas which were previously unavailable or limited, such as indoor locations.

Although a femtocell is a simple plug-and-play device, its location requires pre-approval by the service provider. Moreover, for security reasons, femtocells are usually closed-access (Seidel and Saad, 2010), *i.e.*, pre-authorization is required for a user in order to gain access to it. When an authorized user is within the coverage area of a femtocell, a handover process between a macrocell base station and the femtocell occurs automatically. Similarly, for a user leaving the femtocell coverage area, an opposite handover process would occur, transferring service from the femtocell to the nearest available macrocell.

The introduction of femtocells into the cellular network can be very beneficial to both service providers as well as end users (Chandrasekhar, 2008, Yeh *et al*., 2008, and Claussen *et al.*, 2008). As femtocells are user-friendly devices, consumers require no prior technical knowledge to install a femtocell, thereby making it very convenient for a user to buy, install, and operate a femtocell. Moreover, femtocells are able to plug network coverage gaps between macrocells or microcells. In areas, where users are unable to connect to the cellular network due to gaps in coverage or poor signal strength, the installation of a femtocell, the installation of a femtocell would enable such users to be able to receive a strong and steady signal from the service provider. (This is especially so in urban areas where thick concrete walls and tall high-rise buildings would block off all existing signals from a user to a base station.) Finally, as a femtocell is capable of serving up to sixteen users, each authorized user would be allocated a fairly large portion of the frequency band. This would result in greater quality of service for each individual user, which would otherwise not be possible even in the presence of a macrocell with significant transmitting power due to the large number of supported users spanning the coverage area.

Besides offering various advantages to users, femtocells are also very beneficial to service providers. Through widespread usage of femtocells, large amounts of data would be offloaded from the macrocell. Therefore, in an area with previously high macrocell user density, the signal quality provided would now improve substantially due to the lower macrocell user density within the same coverage area. Service providers could also make use of the fixed location property of femtocells to provide companies with advertising opportunities. Advertisements in the form of text messages could be sent to targeted users in a specific region. For example, femtocells located near or within a shopping mall can be used to broadcast advertisements. Other public announcements such as a flash flood warning due to heavy rain can also be made by making use of available femtocells.

Furthermore, in order to attract users to install femtocells, mobile service providers can offer discounted rates for calls placed under femtocell coverage. This could (potentially) attract more users to purchase and install femtocells in their homes or offices, and consequently relieve some of the macrocell's capacity.

Despite the many advantages offered by femtocells they suffer from a number of operational shortcomings (Chandrasekhar, 2008, Yeh *et al*., 2008, and Claussen *et al.*, 2008). The primary issue is "*interference*". This problem is particularly acute for service providers with single frequency licenses. Femtocells serviced by these providers would operate at the same frequency as existing macrocell base stations, thereby interfering with ongoing transmissions. The interference problem is further exacerbated if users do not undergo the handover process when they enter the femtocell coverage area from the coverage area of a macrocell, resulting in near-far interference that can be detrimental to communications. As aforementioned, an increase in femtocell transmit power registers as interference for macrocell users who in turn react by increasing their transmit power, which results in increased interference for femtocell users. This increasing spiral of interference could potentially lead to a failure in communications.

From previous studies, it is known that smaller base stations such as femtocells must be sufficiently separated from each other and their overall density must be relatively low to ensure improved performance with each additional femtocell installation. The preferred density is roughly below 20% of the coverage of the larger base station. When the femtocell density exceeds this threshold, interference starts to accumulate, resulting in a decrease in capacity gains of additional femtocells or even capacity reductions. However, since the purchase of femtocells is consumer-driven, it is difficult to enforce the necessary distance separation between these units in practice.

One possible method of circumventing this near-far interference problem is to ensure a smooth handover process by the user between the femtocell and macrocell. This is not always feasible as a functional property of a femtocell is that only an authorized user is allowed to access the femtocell. Apart from security issues, if un-authorized users were allowed to connect to a femtocell, authorized users would notice a reduction in their allocated bandwidth. Another way to reduce near-far interference arising due to a femtocell is to deploy DAS antennas. Distributed antenna elements reduce access distance between users and the base station tower resulting in lower transmit power levels for macrocell users, thereby reducing interference.

In this chapter, we focus on developing an analytical framework that provides a way of computing optimal antenna deployments so that the total capacity is maximized in CDMA systems. Such a framework would be critical for service providers who are interested in adopting the use of femtocells within a DAS approach. To the best of our knowledge, no such framework currently exists. In this chapter, we pose and solve the problem of determining optimally distributed DAS deployments for maximizing the total number of admitted users in a CDMA system in the presence of femtocells.

4.2 Problem Definition

We consider a region $\Re$ where the location of one base station is known and an optimal location for a set of DAS antennas is desired. Our goal is to determine these locations so that the uplink user capacity, *i.e.*, the number of simultaneous, single-rate, uplink users, is maximized. For ease of discussion, we describe the location of each DAS antenna within $\Re$ using Cartesian coordinates. In this report, we treat the given base station and each DAS antenna as single-antenna systems. (The model presented in this report can be easily extended to the case of multiple base stations and/or multiple antenna base stations.) Within the region $\Re$, we model traffic requests using a set of demand nodes, indexed by $i \in \{1,...,I\}$. Since user demand is typically a continuous function over the network's geographic region, these demand nodes represent a discretized version of the connection requests within the region. For example, the entire geographic region can be subdivided into *i* smaller subregions and a demand node may represent the average demand (or some percentile demand) of a subregion, located at its center of gravity. For each demand node i, let $(\alpha_i, \beta_i) \in \Re$ denote the location of demand node i, and let h_i denote the number of desired connections, *i.e.*, user terminals, from demand node i. (This approach of modeling user demand via discrete demand nodes, referred to as "hotspots", has been undertaken by several earlier studies on cellular network planning (Amaldi, 2003).) To simplify discussion, let $h_1 \equiv 1, \quad \forall i = 1,...,I$. (Cases of non-unit demand can be easily incorporated by placing multiple demand nodes at, or in the vicinity of, a particular location.)

Here, we assume that:

- At each base, ideal RAKE processing is performed on each user's received signal in order to maximize diversity. RAKE processing is a method used to reduce the signal loss due to multipath fading. It makes use of several receivers each receiving a copy of the same transmission. Multiple transmissions received from different receivers would then be combined to produce a signal of improved quality.
- Each user-base path is "infinitely dispersive", implying that there is a near-infinitude of resolvable, comparable-strength multi-paths on each link.

Given the above two assumptions, the receiver output signal sample for each user will be non-fading just as if the link had a single path of fixed gain. We assume these fixed gains are known and designate the path gains from demand node i to the given base as T_{0i}; from demand node i to a DAS antenna located at position (x_l, y_l) as $T_{li} \equiv T_i(x_l, y_l)$; and from demand node i to the given femtocell as T_{Fi}. With these fixed gains, we can proceed by assuming the additive white Gaussian noise condition for each link.

Ideally, we would like to determine the optimal locations for the DAS antennas over the continuous range of points in $\Re$. However, the difficulty with this goal lies in expressing $T_i(x, y)$ as a continuous function over the range of x and y, as $T_i(x, y)$ depends on several factors such as the distance between demand node i and the DAS antenna at location (x, y), the attenuation due to shadowing, which in turn is related to the nature of the obstacles between the demand node and the DAS antenna (Rappaport, 2002). More realistically, network designers are often limited by local ordinances and regulatory constraints, which limit possible antenna deployments to a discrete set of candidate antenna locations. Moreover, field measurements of $T_i(x, y)$ are available to network designers for a limited set of candidate locations. Therefore, instead of solving the continuous location problem, we focus on finding the optimal DAS antenna locations from amongst a discrete set of points.

Accordingly, let the set of candidate DAS antenna locations be indexed by the set

$l \in \{1,...,L\}$, where each location l has Cartesian coordinates $(x_l, y_l) \in \Re$. We let $l = 0$ denote a given base station located at the origin. With this framework, the signal gain between each user at demand node *i* and the given base can be expressed as $\sqrt{T_{0i}} e^{j\phi_{0i}}$. Similarly, the signal gain between demand node i and an antenna deployed at location (x_l, y_l), $l = 1,...,L$, can be written as $\sqrt{T_{li}} e^{j\phi_{li}}$, and the signal gain between each user at demand node *i* and a given femtocell can be expressed as $\sqrt{T_{Fi}} e^{j\phi_{Fi}}$. The parameters ϕ_{oi}, ϕ_{li} and ϕ_{Fi} are phases of the signal gains from user i to the given base, a DAS antenna at location l, and femtocell F, respectively. In the DAS system studied here, all antennas receive, despread, and RAKE-combine the signal from each admitted user. Therefore, there are multiple output streams (one at the given base and one at each of the distributed antennas), which contain a desired user's signal. Assuming that all admitted user terminals use single-rate CDMA over the same set of frequencies, each of these streams also contains interference from the other supported users plus thermal noise. We assume the output streams at the given base and DAS antenna at location l are weighted by ω_{0i} and ω_{li}, respectively, and then summed over all antennas. Throughout this chapter, we assume that the chosen weights ω_{0i} and ω_{li}, $\forall l = 1,...,L$, reflect the relative path gains between users and system antennas, *i.e.*, an antenna with a higher path gain to an admitted user has a proportionately higher weight than an antenna with a lower path gain to this user.

Then, the objective of the DAS Location Problem (DASLP) under consideration is to determine an optimal set of K locations (chosen from the set of L candidate locations within $\Re$ where DAS antennas may be deployed), such that the total number of user terminals supported is maximized subject to given quality of service (QoS) requirements. These QoS requirements are expressed in terms of the minimum *Signal-to-Interference plus Noise Ratio* **(SINR)** constraints on the uplink. (We model QoS requirements as uplink SINR constraints because in many applications they tend to be more stringent than the corresponding downlink SINR constraints (St-Hilaire *et al.*, 2006)). In doing so, the solution to the DASLP also determines the optimal subset of the i demand nodes that can be admitted into the overall system. Hence, a solution to the DASLP:

- Dictates how the DAS system should be optimally deployed.
- Determines how much of the (user) demand can be supported by the resulting system.

4.3 Model Formulations and Algorithms

4.3.1 An Exact System Model

Given user i's Cartesian coordinates $(\alpha_i, \beta_i) \in \Re$, let $d_{0i} = \sqrt{\alpha_i^2 + \beta_i^2}$ denote the distance from the base station to demand node i. Then, we compute the path gain from user node i to the base station as:

$$T_{0i} = \begin{cases} H_{BS}\left(\dfrac{d_{0i}}{d_{ref}}\right)^{-2} 10^{\chi_1/10}, & d_{0i} \le d_{BS} \\ H_{BS}\left(\dfrac{d_{0i}}{d_{ref}}\right)^{-4} 10^{\chi_1/10}, & d_{0i} > d_{BS} \end{cases}, \tag{4.1}$$

Where, $H_{BS} = 10$, $d_{BS} = 100\text{m}$, $d_{ref} = 1\text{km}$, and χ_1 is a Gaussian random variable with parameters $(\mu_1, \sigma_1^2) = (0, 8)$. Similarly, if d_{ki} represents the distance from demand node i to candidate location l, then the path gain from user i to a DAS antenna at candidate location l is given by,

$$T_{li} = \begin{cases} H_{DAS}\left(\dfrac{d_{li}}{d_{ref}}\right)^{-2} 10^{\chi_2/10}, & d_{li} \le d_{DAS} \\ H_{DAS}\left(\dfrac{d_{li}}{d_{ref}}\right)^{-4} 10^{\chi_2/10}, & d_{li} > d_{DAS} \end{cases}, \tag{4.2}$$

Where, $H_{DAS} = 10$, $d_{DAS} = 100\text{m}$, $d_{ref} = 1\text{km}$, and χ_2 is a Gaussian random variable with parameters $(\mu_2, \sigma_2^2) = (0, 4)$.

Finally, if d_{Fi} represents the distance from demand node i to femtocell location F, then the path gain from user i to a femtocell at candidate location F is given by:

$$T_{Fi} = \begin{cases} H_F \left(\frac{d_{Fi}}{d_{ref}} \right)^{-2} 10^{\chi_3/10}, & d_{Fi} \le d_F \\ H_F \left(\frac{d_{Fi}}{d_{ref}} \right)^{-4} 10^{\chi_3/10}, & d_{Fi} > d_F \end{cases} \tag{4.3}$$

Where, $H_F = 10$, $d_F = 100\text{m}$, $d_{ref} = 1\text{km}$, and χ_3 is a Gaussian random variable with parameters $(\mu_3, \sigma_3^2) = (0,1)$.

Define,

$$z_i = \begin{cases} 1, & \text{demand node } i \text{ is admitted into macrocell} \\ 0, & \text{demand node } i \text{ is not admitted into macrocell} \end{cases}, \quad \forall i = 1,...,\mathrm{I}. \tag{4.4}$$

$$y_i = \begin{cases} 1, & \text{demand node } i \text{ is admitted into femtocell} \\ 0, & \text{demand node } i \text{ is not admitted intofemtocell} \end{cases}, \quad \forall i = 1,...,\mathrm{I}. \tag{4.5}$$

$$q_l = \begin{cases} 1, & \text{antenna is located at candidate location } l \\ 0, & \text{antenna is not located at candidate location } l \end{cases}, \quad \forall i = 1,...,\mathrm{I}. \tag{4.6}$$

We are now ready to formulate the *Distributed Antenna Location with Femtocell Problem* **(DASFP)** of optimally locating a set of K DAS antennas amongst a set of L candidate locations in the presence of a single femtocell.

DASFP: Maximize $\sum_{i=1}^{I} (z_i + y_i)$ (4.7a)

Subject to $q_0 = 1$ (4.7b)

$$\sum_{l=1}^{L} q_l = K, \tag{4.7c}$$

$$\frac{\frac{W}{R} P_i \sum_{l=0}^{L} q_l T_{li}}{\sum_{\substack{j=1 \\ j \ne i}}^{I} z_j \sum_{l=0}^{L} q_l |w_{li}|^2 P_j T_{lj} + \sum_{m=1}^{I} y_m \sum_{l=0}^{L} q_l |w_{li}|^2 P_m T_{lm} + \eta W} \ge \Gamma z_i, \quad \forall i \tag{4.7d}$$

$$\frac{\frac{W}{R} P_i T_{Fi}}{\sum_{\substack{j=1 \\ j \ne i}}^{I} z_j P_j T_{Fi} + \sum_{\substack{i \in N \\ i \ne j}} y_i P_i T_{Fi} + \eta W} \ge \Gamma y_i, \quad \forall i \tag{4.7e}$$

$$|\omega_{li}|^2 = \frac{T_{li}}{\sum_{l=0}^{L} q_l T_{li}}, \quad \forall (i, l) \tag{4.7f}$$

$$\sum_{l=0}^{L'} q_l T_{li} \le T_{Fi} y_i + M(1 - y_i), \forall i \tag{4.7g}$$

$$\sum_{l=0}^{L'} q_l T_{li} \ge T_{Fi} z_i, \forall i \tag{4.7h}$$

$$z_i + y_i \le 1, \quad \forall i \tag{4.7i}$$

$$\eta W \le P_i \le P_{\max}, \quad \forall i \tag{4.7j}$$

$$z_i, y_i, q_l \in \{0, 1\} \tag{4.7k}$$

In this formulation above, the objective function (4.7a) seeks to maximize the number of users admitted into the system, constraint (4.7b) asserts the location of the base station at the origin, constraint (4.7c) restricts the number of DAS antenna locations to be a specified number K, constraints (4.7d) and (4.7e) impose the QoS requirement that all admitted users must meet a minimum SINR of Γ for the macrocell and femtocell, respectively, constraint (4.7f) represents the co-phasing constants, constraints (4.7g) and (4.7h) enforce the necessary criterion for a user to be admitted into the femtocell and macrocell, respectively, constraint (4.7i) states that a user can either be admitted into a macrocell or femtocell but not both, constraint (4.7j) ensures that the power of each admitted user must be at least the power of the AWGN at the receiver, and finally constraint (4.7k) imposes binary restrictions on the $(z, y, q)-$ variables.

Moreover, W denotes the system bandwidth and R is the data rate, which implies that for the CDMA system at hand the spreading gain is W/R, ηW is the AWGN power, $P_{\max}$ is the maximum transmit power for any user, and finally, Γ is the minimum SINR required for each admitted user.

4.3.2 A Reformulated DAS with Femtocell Problem (RDASFP)

For ease of algebraic manipulation and subsequent algorithmic analysis, we define

$K' = \frac{W}{R\Gamma}$, such that $K'+1$ now represents the single cell pole capacity in a CDMA system, and furthermore, for notational convenience, we let $S_i = \frac{P_i}{\eta W}$.

Now, using the substitutions described above yields the following equivalent formulation of **DASFP**, given by:

$$\text{Maximize} \quad \sum_{i=1}^{I}(z_i + y_i) \tag{4.8a}$$

$$\text{Subject to:} \quad q_0 = 1 \tag{4.8b}$$

$$\sum_{l=1}^{L} q_l = K \tag{4.8c}$$

$$K' S_i \sum_{k=1}^{L}\sum_{m=1}^{L}(q_k q_m T_{ki} T_{mi}) > \sum_{\substack{j=1 \\ j\neq i}}^{I}\sum_{l=0}^{L}(z_i s_j z_i q_l T_{li} T_{lj}) + \sum_{m=1}^{I}\sum_{l=0}^{L}(z_i q_l y_m s_m T_{li} T_{lm}) + \sum_{l=0}^{L} q_l z_i T_{li}, \quad \forall i \tag{4.8d}$$

$$K' S_i T_{F_i} > \sum_{\substack{j=1 \\ j\neq i}}^{I} z_j S_j y_i T_{F_i} + \sum_{\substack{m=1 \\ m\neq i}}^{I} y_i y_m S_m T_{F_m} + y_i, \quad \forall i \tag{4.8e}$$

$$\sum_{l=0}^{L} q_l T_{li} \le T_{F_i} y_i + M(1 - y_i), \quad \forall i \tag{4.8f}$$

$$\sum_{l=0}^{L} q_l T_{li} \ge T_{F_i} z_i, \quad \forall i \tag{4.8g}$$

$$z_i + y_i \le 1, \quad \forall i \tag{4.8h}$$

$$0 \le S_i \le S_{\max}, \quad \forall i \tag{4.8i}$$

$$S_i \ge z_i + y_i, \quad \forall i \tag{4.8j}$$

$$z_i, y_i, q_l \in \{0,1\} \tag{4.8k}$$

As stated above, same as expression of (4.7), the objective function (4.8a) aims to maximize the number of users admitted into the system, constraint (4.8b) asserts the location of the base station at the origin, constraint (4.8c) restricts the number of DAS antenna locations to be a specified number K, constraints (4.8d) and (4.8e) impose the QoS requirement that all admitted users must meet a minimum SINR of Γ for the macrocell and femtocell, respectively, constraints (4.8f) and (4.8g) enforce the necessary criterion for a user to be admitted into the femtocell and macrocell, respectively, constraint (4.8h) states that a user can either be admitted into a macrocell or femtocell but not both, constraint (4.8i) ensures that the power of each admitted user must be at least the power of the AWGN at the receiver, and finally constraint (4.8k) imposes binary restrictions on the $(z, y, q)-$ variables.

In order to make DASFP more amenable to algorithmic techniques, we reformulate the problem with the addition of several auxiliary variables. To begin with, let us denote,

$$A_j = z_j S_j, \quad \forall j \in V \tag{4.9}$$

Note this is a nonlinear constraint. In order to generate an equivalent linear formulation, we construct the convex outer-envelope of the function $z_j S_j$ over the bounding rectangle to obtain the reformulated expression, given as follows:

$$A_j \le S_j \tag{4.10a}$$

$$A_j \le S_{\max} z_j \tag{4.10b}$$

$$S_j - S_{\max} + S_{\max} z_j - A_j \le 0 \tag{4.10c}$$

where, in the spirit of RLT (see Sherali and Tuncbilek, 1992 and 1997), constraints (4.10a)-(4.10c) impose $A_j = z_j S_j$ at the optimum.

Akin to the above reformulation, we now can reformulate all the binomial terms appeared in **DASFP**, given as follows:

$$B_{il} = z_i q_l \tag{4.11a}$$

Reformulated as: $B_{il} \le z_i$ (4.11b)

$$B_{il} \le q_l \tag{4.11c}$$

$$-1 + z_i + q_l - B_{il} \le 0 \tag{4.11d}$$

$$C_m = y_m S_m \tag{4.12a}$$

Reformulated as: $C_m \le S_m$ (4.12b)

$$C_m \le S_{\max} z_m \tag{4.12c}$$

$$S_m - S_{\max} + S_{\max} z_m - C_m \le 0 \tag{4.12d}$$

$$D_{km} = q_k q_m \tag{4.13a}$$

Reformulated as: $D_{km} \le q_m$ (4.13b)

$$D_{km} \le q_k \tag{4.13c}$$

$$-1 + q_k + q_m - D_{km} \le 0 \tag{4.13d}$$

$$G_{ji} = z_j S_j y_i = A_j y_i \tag{4.14a}$$

Reformulated as: $G_{ji} \le A_j$ (4.14b)

$$G_{ji} \le S_{\max} y_i \tag{4.14c}$$

$$A_j - S_{\max} + S_{\max} y_i - G_{ji} \le 0 \tag{4.14d}$$

$$H_{mi} = y_i y_m S_m = C_m y_i \tag{4.15a}$$

Reformulated as: $H_{mil} \le C_m$ (4.15b)

$$H_{mi} \le S_{\max} y_i \tag{4.15c}$$

$$C_m - S_{\max} + S_{\max} y_i - H_{mi} \le 0 \tag{4.15d}$$

Next, to account for the trinomial terms appearing in the model, define,

$$E_{jil} = z_j S_j z_i q_l = A_j B_{il} \tag{4.16a}$$

Reformulated as: $E_{jil} \le A_j$ (4.16b)

$$E_{jil} \le S_{\max} B_{il} \tag{4.16c}$$

$$A_j - S_{\max} + S_{\max} B_{il} - E_{jil} \le 0 \tag{4.16d}$$

$$F_{mil} = y_m S_m z_i q_l = C_m B_{il} \tag{4.17a}$$

Reformulated as: $F_{mil} \le C_m$ (4.17b)

$$F_{mil} \le S_{\max} B_{il} \tag{4.17c}$$

$$C_m - S_{\max} + S_{\max} B_{il} - F_{mil} \le 0 \tag{4.17d}$$

$$O_{ikm} = S_i D_{km} = S_i q_k q_m \tag{4.18a}$$

Reformulated as: $O_{ikm} \le S_i$ (4.18b)

$$O_{ikm} \le S_{\max} D_{km} \tag{4.18c}$$

$$S_i - S_{\max} + S_{\max} D_{km} - O_{ikm} \le 0 \tag{4.18d}$$

Now, we can rewrite the formations and obtain the *Reformulated DAS Location with Femtocell Problem* (**RDASFP**) using the substitutions described above.

RDASFP: Maximize $\sum_{i=1}^{I} (z_i + y_i)$ (4.19a)

Subject to: $q_0 = 1$ (4.19b)

$$\sum_{l=1}^{L} q_l = K \tag{4.19c}$$

$$\left.\begin{array}{l} K' \sum_{k=1}^{L} \sum_{m=1}^{L} O_{ikm} T_{ki} T_{mi} > \sum_{\substack{j=1 \\ j \ne i}}^{I} \sum_{l=0}^{L} E_{jil} T_{li} T_{lj} + \\ \sum_{m=1}^{I} \sum_{l=0}^{L} F_{mil} T_{li} T_{lm} + \sum_{l=0}^{L} B_{il} T_{li} \end{array}\right. , \quad \forall i \tag{4.19d}$$

$$K' S_i T_{F_i} > \sum_{\substack{j=1 \\ j \ne i}}^{I} G_{ji} T_{F_i} + \sum_{\substack{m=1 \\ m \ne i}}^{I} H_{mi} T_{F_m} + y_i, \quad \forall i \tag{4.19e}$$

$$\sum_{l=0}^{L} q_l T_{li} \le T_{F_i} y_i + M(1 - y_i), \quad \forall i \tag{4.19f}$$

$$\sum_{l=0}^{L} q_l T_{li} \ge T_{F_i} z_i, \quad \forall i \tag{4.19g}$$

$$\left.\begin{array}{c} A_j \le S_j \\ A_j \le S_{\max} z_j \\ S_j - S_{\max} + S_{\max} z_j - A_j \le 0 \end{array}\right\}, \quad \forall j \tag{4.19h}$$

$$\left.\begin{aligned} B_{il} &\le z_i \\ B_{il} &\le q_l \\ -1 + z_i + q_l - B_{il} &\le 0 \end{aligned}\right\}, \quad \forall (i, l) \tag{4.19i}$$

$$\left.\begin{aligned} C_m &\le S_m \\ C_m &\le S_{\max} z_m \\ S_m - S_{\max} + S_{\max} z_m - C_m &\le 0 \end{aligned}\right\}, \quad \forall m \tag{4.19j}$$

$$\left.\begin{aligned} D_{km} &\le q_m \\ D_{km} &\le q_k \\ -1 + q_k + q_m - D_{km} &\le 0 \end{aligned}\right\}, \quad \forall (k, m) \tag{4.19k}$$

$$\left.\begin{aligned} E_{jil} &\le A_j \\ E_{jil} &\le S_{\max} B_{il} \\ A_j - S_{\max} + S_{\max} B_{il} - E_{jil} &\le 0 \end{aligned}\right\}, \quad \forall (i, j, l) \tag{4.19l}$$

$$\left.\begin{aligned} F_{mil} &\le C_m \\ F_{mil} &\le S_{\max} B_{il} \\ C_m - S_{\max} + S_{\max} B_{il} - F_{mil} &\le 0 \end{aligned}\right\}, \quad \forall (i, m, l) \tag{4.19m}$$

$$\left.\begin{aligned} G_{ji} &\le A_j \\ G_{ji} &\le S_{\max} y_i \\ A_j - S_{\max} + S_{\max} y_i - G_{ji} &\le 0 \end{aligned}\right\}, \quad \forall (i, j) \tag{4.19n}$$

$$\left.\begin{aligned} H_{mi} &\le C_m \\ H_{mi} &\le S_{\max} y_i \\ C_m - S_{\max} + S_{\max} y_i - H_{mi} &\le 0 \end{aligned}\right\}, \quad \forall (i, m) \tag{4.19o}$$

$$\left.\begin{aligned} O_{ikm} &\le S_i \\ O_{ikm} &\le S_{\max} D_{km} \\ S_i - S_{\max} + S_{\max} D_{km} - O_{ikm} &\le 0 \end{aligned}\right\}, \quad \forall (i, k, m) \tag{4.19p}$$

$$z_i + y_i \le 1, \tag{4.19q}$$

$$0 \le S_i \le S_{\max} \tag{4.19r}$$

$$S_i \ge z_i + y_i \tag{4.19s}$$

$$0 \le A_j, C_m, E_{jil}, F_{mil}, G_{ji}, H_{mi}, S_i \le S_{\max} \tag{4.19t}$$

$$z_i, y_i, q_l, B_{il}, D_{km} \in \{0,1\} \tag{4.19u}$$

In this formulation above, the objective function (4.19a) seeks to maximize the number of users admitted into the system, constraint (4.19b) asserts the location of the base station at the origin, constraint (4.19c) restricts the number of DAS antenna locations to be a specified number K, constraints (4.19d) and (4.19e) represent the linearized version of the QoS requirement under the substitutions (4.9)-(4.18), constraints (4.19f) and (4.19g) enforce the necessary criterion for a user to be admitted into the femtocell and macrocell, respectively; constraints (4.19h) and (4.19p) enforce the condition that any *feasible* solution to **RDASFP** the auxiliary variables faithfully reproduce their corresponding nonlinear products (4.9)-(4.18), respectively; constraint (4.7q) states that a user can either be admitted into a macrocell or femtocell but not both, constraint (4.7r) ensures that the power of each admitted user must be at least the power of the AWGN at the receiver, and finally constraint (4.7k) imposes binary restrictions on the $(z, y, q, B, D)-$variables. Thus, by the addition of a *polynomial* number of variables and constraints, we have obtained an equivalent *higher-dimensional mixed-integer 0-1 linear programming representation* of the original nonconvex binary multilinear program.

The process adopted in our approach of linearizing Problem **DASFP** via the addition of new variables (given in (4.10a)-(4.10c)), and constraints (4.11a)-(4.18d) is based on the principle of *disjunctive programming*. We note here that there are several other schemes, such as quadrification, described in Shor (1990), which can be used to reduce Problem DASFP to a quadratic programming problem. This quadratic program can then be subsequently linearized using the process described in our approach. Another alternative would be to directly treat Problem DASFP as a *polynomial programming* problem, and use the methods described in Sherali and Tuncbilek (1992, 1997 and 1998), to derive an equivalent LP representation. However, these schemes would lead to the addition of variables, of far greater orders of magnitude, which prohibits the use of these methods for large-scale instances.

For the sake of convenience, we shall denote the set of variables in **RDASFP**, with obvious vector notation, as

$$\xi \equiv (A, B, C, D, E, F, G, H, S, z, y, q). \qquad (4.20)$$

Then, the following results are evident, where $\nu[\bullet]$ represents the optimal objective function value of the Problem $[\bullet]$

LEMMA 1. *Let RDASFP-LP represent the linear programming (LP) relaxation of RDASFP. Then* $\nu[\text{RDASFP-LP}]$ *provides an upper bound on* $\nu[\text{RDASFP}]$ *and furthermore, if* $\bar{\xi} \equiv \left(\bar{A}, \bar{B}, \bar{C}, \bar{D}, \bar{E}, \bar{F}, \bar{G}, \bar{H}, \bar{S}, \bar{z}, \bar{y}, \bar{q}\right)$ *solves RDASFP-LP and satisfies the constraints (4.19h) – (4.19p), then it is optimal to DASFP*.

Proof. Obvious from construction of *RDASFP*. □

PROPOSITION 1. *Given a solution* $\bar{\xi} \equiv \left(\bar{A}, \bar{B}, \bar{C}, \bar{D}, \bar{E}, \bar{F}, \bar{G}, \bar{H}, \bar{S}, \bar{z}, \bar{y}, \bar{q}\right)$ *to RLDASFP-LP, a feasible solution to DASFP can be obtained via the following heuristic*.

4.3.3 A Tightened Reformulation System Model

In order to achieve a more realistic comparison and to develop tighter (smaller) upper bounds, we augment Problem **RDASFP** using some valid inequalities, such as:

$$\sum_{l=0}^{L} B_{il} T_{li} \geq T_{F_i} z_i, \quad \forall i, \tag{4.23}$$

$$\sum_{l=0}^{L} q_l T_{li} - \sum_{l=0}^{L} B_{il} T_{li} \geq 0, \quad \forall i. \tag{4.24}$$

Constraints (4.23) enforce the necessary criterion for a user to be admitted into the macrocell, not the femtocell. Constraint (4.24) states that the accumulated path gains for user i can get from all the candidate DAS antenna locations greater than the path gains that user i get from the macrocells. Constraint (4.23) and constraint (4.24) enforce a tighter condition comparing to constraint (4.19g), which result in a tighter upper bound.

4.4 Computational Results

In our computations, we define the region $\Re$ to be a symmetrical square of area one-square kilometer centered at the origin, *i.e.*,

$$\Re \equiv \{(x, y) : (x, y) \in [-0.5, 0.5] \times [-0.5, 0.5]\}$$

In this region, the base station (BS) is located at the origin, and the Cartesian coordinates for four points were randomly generated via a uniform distribution as the set of possible candidate DAS antenna locations. The coordinates of the femtocell were also randomly generated via a uniform distribution. Using these locations, the values of the path gains (namely the $T_{0i}, T_{li}, T_{Fi}-$values) from user i to the base station, macrocells, and femtocell were generated using equations (4.1) – (4.3), respectively. Next, we set $W = 128$, $R = 1$, and $\Gamma = 5$, which results in $K' = \frac{128}{(1)\times(5)} = 25.6$. Finally, let $S_{max} = 10$, which limits the transmit power constraints (4.7j) to be reformulated as $0 \le S_i \le S_{\max}$. Finally, two different demand scenarios were generated by varying the number of users in the region as summarized below:

Scenario 1: 75 users uniformly distributed across $\Re$.

Scenario 2: 100 users uniformly distributed across $\Re$.

Using the above defined data generation parameters, the proposed optimization models were implemented in the *General Algebraic Modeling System* **(GAMS)** in concert with the optimization solver SBB.

4.4.1Results for Scenario 1

In Scenario 1, we let the numbers of users in $i = 75$. In this case, all 75 users were able to receive service, either from the femtocell, or the macrocell, as shown in Table 22 below. Moreover, in this scenario, 15 out of the 75 users were serviced through the femtocell, identified as user numbers 4, 9, 10, 17, 26, 28, 31, 33, 38, 43, 50, 52, 62, 68 and 70, and the other users receive service from the macrocell, *i.e.* either through the DAS antennas or directly from the base station.

Moreover, from Figure 12, we can see that the four possible DAS antenna locations are scattered fairly evenly across the cell, with the location of the base station at the origin (symbolized by the red circle), and the femtocell location is shown using a red square marker. The triangular markers symbolize the selected DAS antennas while the red-cross markers are indicative of the non-selected DAS antenna locations. Moreover, the diamond markers show the location of the randomly generated users in the cell. Among the 4 possible DAS antenna locations, one possible reason the antenna at $(-0.39, 0.42)$ was not chosen is that it is located near the fringe of the cell and would not be able to provide adequate coverage for the cell. Another observation is that the users served by the femtocell are tightly grouped and are nearest to the femtocell.

Figure 12: A scatter plot illustration of scenario 1

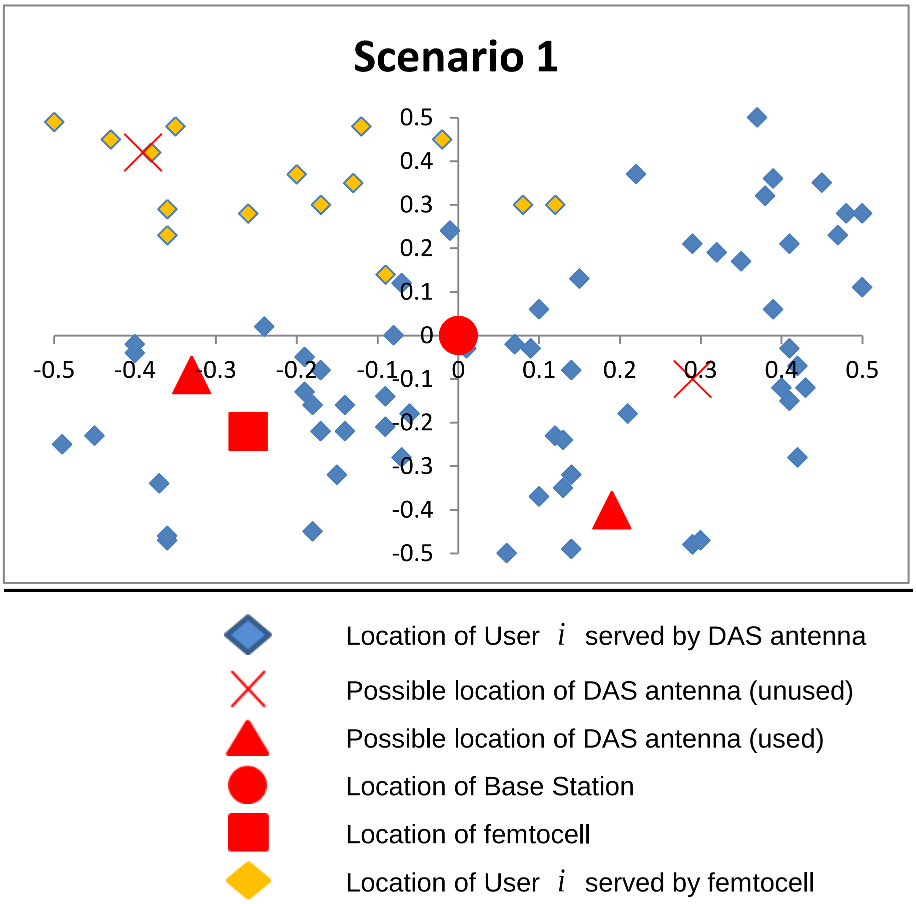


Table 22: Breakdown of result for scenario 1

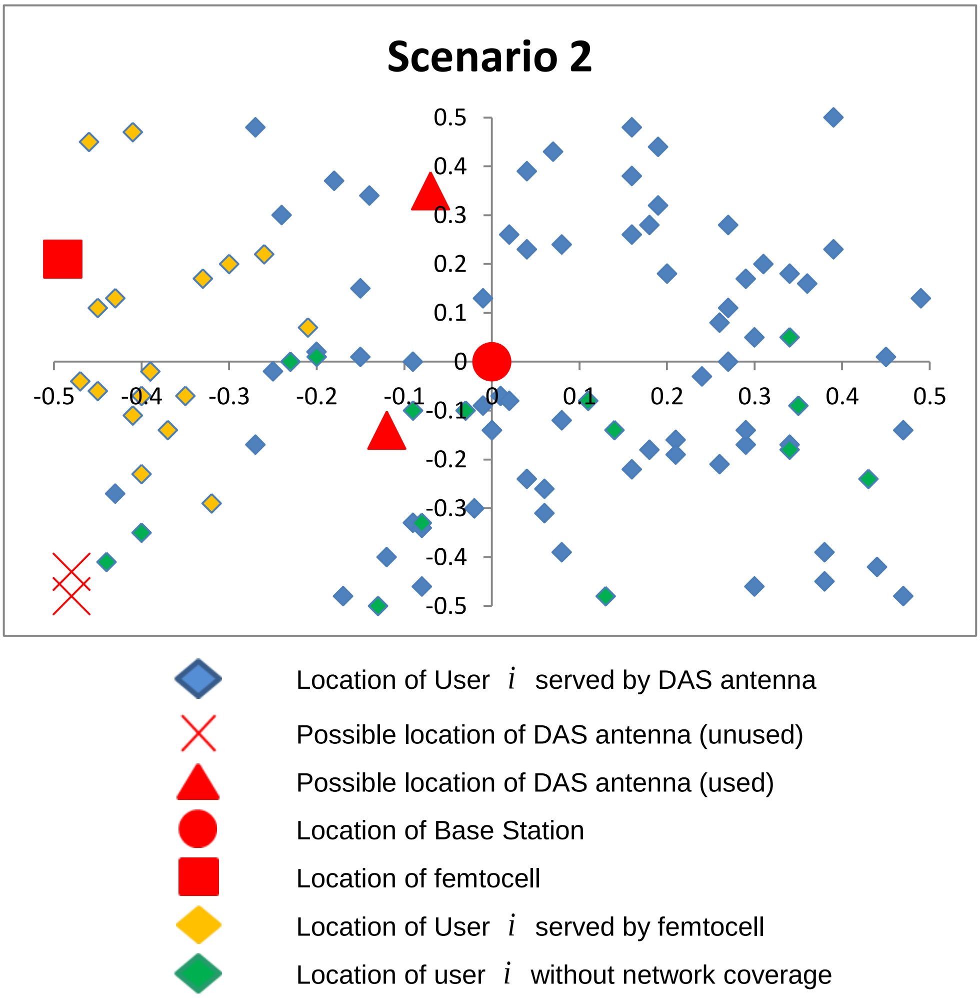
User	MacroCell	FemtoCell	User	MacroCell	FemtoCell	User	MacroCell	FemtoCell
1	1		26		1	51	1	
2	1		27	1		52		1
3	1		28		1	53	1	
4		1	29	1		54	1	
5	1		30	1		55	1	
6	1		31	1	1	56	1	
7	1		32	1		57	1	
8	1		33		1	58	1	
9		1	34	1		59	1	
10		1	35	1		60	1	
11	1		36	1		61	1	
12	1		37	1		62		1
13	1		38		1	63	1	
14	1		39	1		64	1	
15	1		40	1		65	1	
16	1		41	1		66	1	
17		1	42	1		67	1	
18	1		43		1	68		1
19	1		44	1		69	1	
20	1		45	1		70		1
21	1		46	1		71	1	
22	1		47	1		72	1	
23	1		48	1		73	1	
24	1		49	1		74	1	
25	1		50	1	1	75	1	

4.4.2 Results for Scenario 2

For Scenario 2, we let the number of users $i = 100$, and in this case, from Figure 13, we can see that the four possible locations for the DAS antennas were randomly generated towards the left of the cell. Once again, the location of the base station is at the origin (symbolized by the red circle), the femtocell location is shown using a red square marker, and the triangular markers symbolize the selected DAS antennas while the red-cross markers represent the non-selected DAS antenna locations. Finally, the diamond markers show the location of the randomly generated users in the cell.

The two optimized locations for the DAS antennas are both situated near the center of the cell. In addition, as the location of the 2 DAS antennas both lie towards the left of the cell, the coverage across the cell is not uniformly distributed, which leads to a maximum of 85% of users who can be supported simultaneously at any given time in this network. Furthermore, it notes that users without network coverage do not follow any fixed (or observable) trend. Some users who are located at a large distance away from the DAS antenna or base station or femtocell are able to receive network coverage while some users who are at a nearer distance are left unserviced. This can be attributed to the near-far interference problem alluded to earlier.

Figure 13: A scatter plot illustration of scenario 2



In contrast to the first scenario, from the results reported in Table 23, it can be seen that only 85% of users were able to receive service from either the macrocell or the femtocell, out of which 17% of users were serviced by the femtocell. These users are user number 4, 19, 22, 32, 33, 38, 39, 60, 67, 70, 76, 78, 80, 82, 87, 89 and 91. (Note that user numbers 8, 21, 24, 26, 30, 40, 46, 49, 56, 61, 62, 85, 86, 94 and 99 are unable to receive service from the cellular network.) Finally, comparing across scenarios 1 and 2, it is evident that the service levels in a network are dependent on both the number of users as well as the locations of the DAS antennas and femtocells present in the network.

Table 23: Breakdown of result for scenario 2

User	MacroCell	FemtoCell	No Service	User	MacroCell	FemtoCell	No Service
1	1			51	1		
2	1			52	1		
3	1			53	1		
4		1		54	1		
5	1			55	1		
6	1			56			1
7	1			57	1		
8			1	58	1		
9	1			59	1		
10	1			60		1	
11	1			61			1
12	1			62			1
13	1			63	1		
14	1			64	1		
15	1			65	1		
16	1			66	1		
17	1			67		1	
18	1			68	1		
19		1		69	1		
20	1			70		1	
21			1	71	1		
22		1		72	1		
23	1			73	1		
24			1	74	1		
25	1			75	1		

26			1	76		1	
27	1			77	1		
28	1			78		1	
29	1			79	1		
30			1	80		1	
31	1			81	1		
32		1		82		1	
33		1		83	1		
34	1			84	1		
35	1			85			1
36	1			86			1
37	1			87		1	
38		1		88	1		
39		1		89		1	
40			1	90	1		
41	1			91	1	1	
42	1			92	1		
43	1			93	1		
44	1			94			1
45	1			95	1		
46			1	96	1		
47	1			97	1		
48	1			98	1		
49			1	99			1
50	1			100	1		

4.5 Extensions and Conclusions

In this chapter, we extended the methodology developed in Chapter 3 to a CDMA networks in the presense of femtocells. The "femtocell" is a new and upcoming wireless techonoly which can improve the coverage and provide high quality service in a cost effective manners. We posed and solved the problem of determining a set of distributed DAS antennas for maximizing the total number of users (user capacity) in a CDMA system such that each admitted user satisfies the minimum SINR requirement while respecting the maximum transit power constraint in the presense of femtocells. This complex problem was formulated as a mixed-integer 0-1 multilinnear program, where the largest nonlinear (and nonconvex) term is a quartic-degree polynomial. We reformulated this nonconvex problem using auxiliary variables and constraints to yield and equivalent higher dimensional mixed-integer 0-1 linear programming problem. In order to achieve a more realistic comparison and to develop tighter upper bounds, we also augment the problem using some valid inequalities. The computational results for differen scenarios demonstrated the efficiency of the proposed algorithm, and the global optimum was determined for all the test instances.

From an algorithmic and computational point of view, a rigorous analytical framework, such as the one developed in this paper, is very useful for service providers opting for a DAS-based network solution. The exact global optimization framework, accompanied by a specialized linearization strategy and search heuristic techniques, presented in our work leads to tight lower and upper bounds at every iteration, thereby leading to the desired algorithmic convergence. Moreover, a generic approach such as the one prescribed in this chapter can form the basis for addressing a host of problems arising in cellular network location problems.

As mentioned previously in the limitations, there are a larger number of possible combinations of number of antennas. As such, perhaps more work could be done to try out the different scenarios. For example, future work could involve the inclusion of two or more femtocells within a cell. It could even involve fixing a number of femtocells for public usage while having a number of femtocells which are closed access.

Chapter 5

Conclusions

While most of the research efforts thus far involving fractional programs have focused on solving various classes of the single-ratio fractional programs, or multi-ratio cases involving linear fractional functions. However, fractional programming problems become substantially more difficult as the number of ratios in the objective function increases, and this complexity is further amplified when the involved terms are higher-order polynomials. While some classes of multiple-ratio problems can be solved to optimality by recasting them as equivalent *0-1 mixed-integer programming problems*, and embedding them within a tailored branch-and-bound algorithm, there is yet significant scope to devise specialized algorithms for solving the generic class of sum of ratios fractional programs, particularly for the case where the involved functions are higher-degree polynomials.

In this dissertation, we derived globally optimail algorithms for three specialized fractional programming problems arising in the context of the independent set problem and management applications in wireless communications.

We investigated continuous optimization approaches, notably fractional programming methods, to determine the stability number (or independence number) of a graph. While, traditionally, this problem has been solved using integer programming methods, but these methods have been known to suffer from a number of shortcomings, in addition to the complexity of dealing with discrete variables. In this context, we defined a new class of *vertex sets*, and we ultilized the structure of these vertex sets to derive explicit characterizations of the number of alternate optima present in both discrete and continuous formulations. The computational results showed that these vertex sets enabled a simple, yet powerful, construction procedure to efficiently determine maximal independent sets.

We also developed a global optimization algorithm (reformulated fractional programming) to solve the FP formulation, and we demonstrated that this continuous approach stays on par with the 0-1 discrete formulation with respect to various performance metrics. We also presented a fast LP-rounding based heuristic to determine tight lower bounds on the independence number. Putting these theoretical contributions together, as seen in our numerical experiments, we showed that the computational time required per optimal solution is comparable and in some instances lower for Problem FP as compared to Problem MIS (as the number of alternate optima for Problem FP is significantly greater when compared to Problem MIS).

The second portion of this thesis dealt with real-world applications arising in the context of cellular network design. We posed and solved the problem of determining a set of distributed DAS antennas for maximizing the total number of users (user capacity) in a CDMA system such that each admitted user satisfies the minimum SINR requirement while respecting the maximum transit power constraint. This complex problem was formulated as a mixed-integer 0-1 multilinnear program, where the largest nonlinear (and nonconvex) term is a quartic-degree polynomial. We presented a set of (exact and approximate) mathematical models and algorithms for this problem. This highly nonlinear, nonconvex problem is reformulated to yield a tight mixed-integer 0-1 linear programming representation via the addition of several auxiliary variables and constraints. We designed a tailored branch-and-bound algorithm, in conjunction with an intelligent heuristic algorithm to solve the linearized problem. And the computational results showed that the the linearized models in conjunction with the proposed algorithm clearly outperformed the original nonlinear model.

Finally, we extended the above developed methodology for optimally locating DAS antennas in CDMA cellular networks in the presence of femtocells. An often recurring complaint among consumers concerns the imperfect coverage of current cellular networks, and this problem is particularly acute in indoor locations leading to an increased probability of signal loss. In such situations, a new and upcoming cellular network technology known as "*femtocells*" provides a promising option for cellular companies to improve their coverage, and provide high quality services in a cost effective manner. As femtocells are user-friendly devices, consumers require no prior technical knowledge to install a femtocell, thereby making it very convenient for a user to buy, install, and operate a femtocell.

We developed an analytical framework that provided a way of computing optimal DAS deployments so that the total capacity is maximized in the CDMA cellular network. Such a framework would be critical for service providers who are interested in adopting the use of femtocells. The results showed that the location of the DAS antennas would affect the number of users receiving service from the network providers. The optimum locations for the antennas with the most number of users were found in both scenarios.

In conclusion, this thesis provides an algorithmic framework for deriving globally optimal solutions for the class of fractional programming problems, and this approach is validated by solving various practical problems arising in engineering and management applications.

References

[1] Abello, J., Butenko, S., Pardalos, P.M. and Resende M.G.C. (2001), Finding independent sets in a graph using continuous multivariable polynomial formulations, *Journal of Global Optimization* 21, 111–137.

[2] Ageev, D.V. (1935), Bases of the Theory of Linear Selection. Code Demultiplexing, In *Proceedings of the Leningrad Experimental Institute of Communication* 3, 35.

[3] Almogy, Y. and Levin, O. (1970), Parametric analysis of a multistage stochastic shipping problem, *Operational Research* 69, 659-370.

[4] Amaldi, E., Capone, A. and Malucelli, F. (2001, July), Discrete models and algorithms for the capacitated location problems arising in UMTS network planning. In *Proceedings of the 5th international workshop on Discrete algorithms and methods for mobile computing and communications*, ACM, 1-8.

[5] Amaldi, E., Capone, A. and Malucelli, F. (2003), Planning UMTS base station location: Optimization models with power control and algorithms. *IEEE Transactions* on *Wireless Communications*, 2(5), 939-952.

[6] Amaldi, E., Capone, A., Malucelli, F. and Mannino, C. (2006), Optimization problems and models for planning cellular networks. In *Handbook of optimization in telecommunications* 917-939. Springer US.

[7] Anstreicher, K.M. (1986), A monotonic projective algorithm for fractional linear programming, *Algorithmica* 1(4): 483– 498.

[8] Asmussen, S. (1987), *Applied probability and queues*, John Wiley & Sons, New York, USA.

[9] AT&T chooses Raleigh, Charlotte for 3G MicroCell trial [Online] September 28, 2009. Available from: http://www.LocalTechWire.com

[10] Avriel, M., Diewert, W.E., Schaible, S., and Zang, I. (1988), *Generalized Concavity*, Plenum, New York, USA.

[11] Balas, E. and Yu, C. (1986), Finding a maximum clique in an arbitrary graph. *SIAM Journal of computing* 15, 1054-1068.

[12] Balasundaram, B. and Butenko, S. (2005), *Constructing test functions for global optimization using continuous formulations of graph problems*, Optimization Methods and Software 20, 439-452.

[13] Balasundaram, B. and Butenko, S. (2006), On a polynomial fractional formulation for independence number of a graph, J*ournal of Global Optimization* 35, 405–421.

[14] Bednarek, A.R. and Taulbee, O.E. (1996), On maximal chains. *Romanian Journal of Pure and Applied Mathematics* 11, 23-25.

[15] Barros, A.I., Frenk, J.B.G., Schaible, S. and Zhang, S. (1996a), A new algorithm for generalized fractional programs. *Mathematical programming* 72(2), 147-175.

[16] Barros, A.I., Frenk, J.B.G., Schaible, S. and Zhang, S. (1996b), Using duality to solve generalized fractional programming problems. *Journal of Global Optimization* 8(2), 139-170.

[17] Barros, A.I., Frenk, J.B.G. and Gromicho, J. (1997), Fractional location problems, *Location Science* 5(1), 47–58.

[18] Barros, A.I., Dekker, R., Frenk, J.B.G. and van Weeren, S. (1997), Optimizing a general optimal replacement model by fractional programming techniques, *Journal of Global Optimization* 10, 405– 423.

[19] Barros, A.I. (1998), *Discrete and fractional programming*, Kluwer, Dordrecht, The Netherlands.

[20] Bector, C.R., Chandra, S. and Singh, C. (1990), Duality in multiobjective fractional programming, In *Generalized Convexity and Fractional Programming with Economic Applications*, *Lecturer Notes Economics and Math Systems* 345: 232–241, Springer, Berlin.

[21] Benoist, J. (1998), Connectedness of the efficient set for strictly quasiconcave sets for strictly quasiconcave sets, *Journal of Optimization Theory and Applications* 96: 627–654.

[22] Bereanu, B. (1965), Decision regions and minimum risk solutions in linear programming, *In Colloquium of Applications of Mathematics to Economics, Hungarian Academy of Science*, *Hungary*, *Budapest* 37-42.

[23] Bernard, J.C. and Ferland, J.A. (1989), Convergence of interval-type algorithms for generalized fractional programming, *Mathematical programming* 43(1-3), 349-363.

[24] Bomze, I.M. (1997), Evolution towards the maximum clique, *Journal of Global Optimization* 10(2), 143-164.

[25] Boncompte, M. and Martinez-Legaz, J.E. (1991), Fractional programming by lower subdifferentiability techniques, *Journal of Optimization Theory and Applications* 68(1), 95–116.

[26] Bonner, R.E. (1963), On some clustering techniques, *IBM Journal of Research and Development* 8, 22-32.

[27] Bourjolly, J.M., Laporte, G. and Mercure, H. (1997), A combinatorial column generation algorithm for the maximum stable set problem, *Operations Research Letters* 20(1), 21 – 29.

[28] Bron, C. and Kerbosch, J. (1973), Algorithm 457, Finding all cliques on an undirected graph, *Communications of ACM* 16, 575–577.

[29] Busygin, S., Butenko, S. and Pardalos, P.M. (2002), A heuristic for the maximum independent set problem based on optimization of a quadratic over a sphere, *Journal of Combinatorial Optimization* 6, 287–297.

[30] Busygin, S., Prokopyev, O.A. and Pardalos, P.M. (2005), Feature selection for consistent biclustering via fractional 0–1 programming, *Journal of Combinatorial Optimization* 10(1), 7-21.

[31] Cambini, A., Castagnoli, E., Martein, L., Mazzoleni, P. and Schaible, S. (Eds.) (1990), Generalized convexity and fractional programming with economic applications, *Proceedings of the 3rd International Workshop at Pisa, Italy, Lecture Notes Economics and Math Systems* 345, Springer, Berlin.

[32] Cambini, A., Schaible, S. and Sodini, C. (1993), Parametric linear fractional programming for an unbounded feasible region, *Journal of Global Optimization*: 157–169.

[33] Cambini, A. and Martein, L. (1998), Generalized concavity in multiobjective programming, In *Generalized Convexity, Generalized Monotonicity: recent results* 453-467, Springer, US.

[34] Carraghan. R. and Pardalos. P. (1990), An exact algorithm for the maximum clique problem, *Operations Research Letters* 9, 375–382.

[35] Chandrasekhar, V., Andrews, J.G. and Gatherer, A. (2008), Femtocell networks: a survey, *Communications Magazine, IEEE* 46(9), 59-67.

[36] Charnes, A. and Cooper, W.W. (1962), Programming with linear fractional functionals, *Naval Research Logistics Quarterly* 9: 181– 186.

[37] Charnes, A. and Cooper, W.W. (1963), Deterministic equivalents for optimizing and satisficing under chance constraints, *Operations Research* 11, 18–39.

[38] Charnes, A., Cooper, W.W. and Rhodes, E. (1978), Measuring the efficiency of decision making units, *European Journal of Operational Research* 2, 429– 444.

[39] Charnes, A., Cooper, W.W., Lewin, A.Y. and Seiford, L.M. (Eds.) (1994), *Data Envelopment Analysis: Theory, Methodology and Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands.

[40] Chen, H.J., Schaible, S. and Sheu, R.L. (2009). Generic algorithm for generalized fractional programming, *Journal of optimization theory and applications* 141(1), 93-105.

[41] Chvátal, V. (1977), Determining the stability number of a graph, *SIAM Journal of computing* 6, 643-662.

[42] Claussen, H. (2007, September), Performance of macro-and co-channel femtocells in a hierarchical cell structure. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on* 1-5.

[43] Claussen, H., Ho, L.T. and Samuel, L.G. (2008), An overview of the femtocell concept, *Bell Labs Technical Journal* 13(1), 221-245.

[44] Craven, B.D. (1988), Fractional programming, *Sigma Series Applied Mathematics* 4, Heldermann, Berlin.

[45] Crouzeix, J.P., Ferland, J.A. and Schaible, S. (1985), An algorithm for generalized fractional programs, *Journal of Optimization Theory and Applications* 47(1), 35-49.

[46] Crouzeix, J.P. and Ferland, J.A. (1991), Algorithms for generalized fractional Programming, *Mathematical Programming* 52, 191-207.

[47] Crouzeix, J.P., Martinez-Legaz, J.E. and Volle, M. (Eds.) (1998), Generalized convexity, generalized monotonicity: Recent results, *Proceedings of the 5th International Symposium*, Kluwer Academic Publishers, Dordrecht, The Netherlands.

[48] Dai, L., Zhou, S. and Yao, Y. (2005), Capacity analysis in CDMA distributed antenna systems, *IEEE Transactions on Wireless Communications,* 4(6), 2613-2620.

[49] Dai, L. (2008, March), Distributed antenna system: Performance analysis in multi-user scenario, In *Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on* 85-89. IEEE.

[50] Desler, J.F. and Hakimi, S.L. (1970), On finding a maximum internally stable set of a graph, *Proceedings Of Fourth Annual Princeton Conference on information Science and Systems* 4, 459-462, Princeton, NJ.

[51] Dinkelbach, W. (1967), On nonlinear fractional programming, *Management Science* 13, 492–498.

[52] Duong, P.C. (1987), Finding the global extremum of a polynomial function, In: *Essays on Nonlinear Analysis and Optimization Problems, Institute of Mathematics, Hanoi* 111–120.

[53] Ebenegger, C., Hammer, P.L. and de Werra D. (1984), Pseudo-Boolean functions and stability of graph, *Annals of Discrete Mathematics* 19, 83-98.

[54] Eisenblätter, A., Fügenschuh, A., Geerdes, H.F., Junglas, D., Koch, T., and Martin, A. (2004, March), Integer programming methods for UMTS radio network planning, In *Proceedings of WiOpt'04: Modeling and Optimization in Mobile.*

[55] Elhedhli, S. (2005), Exact solution of a class of nonlinear knapsack problems, *Operations Research Letters* 33(6), 615-624.

[56] Falk, J.E. (1969), Maximization of signal-to-noise ratio in an optical filter, *SIAM Journal on Applied Mathematics* 7, 582–592.

[57] Femtocell Forums (Online) Available: http://www.smallcellforum.org/

[58] Frenk, J.B.G., Dekker, R. and Kleijn, M. (1997), On the marginal cost approach in maintenance, *Journal of Optimization Theory and Applications* 94(3), 771– 781.

[59] Frenk, J.B.G. and Kleijn, M.J. (1998), On regenerative processes and inventory control, *Pure and Applied Mathematics* 9, 61–94.

[60] Freund, R.W. and Jarre, F. (1994), An interior-point method for fractional programs with convex constraints, *Mathematical Programming* 67(3), 407–440.

[61] Friden, C., Hertz, A. and de Werra D. (1990), TABARIS, An exact algorithm based on tabu search for finding a maximum independent set in a graph, *Computers and Operations Research* 17, 437-445.

[62] Galota, M., Glaßer, C., Reith, S. and Vollmer, H. (2001, July), A polynomial-time approximation scheme for base station positioning in UMTS networks, In *Proceedings of the 5th international workshop on Discrete algorithms and methods for mobile computing and communications*, ACM 52-59.

[63] Gardiner, E.J., Artymiuk, P.J. and Willett. P. (1998), Clique-detection algorithms for matching three-dimensional molecular structures, *Journal of Molecular Graphics and Modeling* 15, 245–253.

[64] Garey, M. and Johnson, D. (1979), *Computers and Intractability: A Guide to the Theory of NP-completeness*, Freeman, San Francisco, CA.

[65] Gilmore, P.C. and Gomory, R.E. (1963), A linear programming approach to the stock cutting problem – Part II, *Operations Research* 11, 863–888.

[66] Granot, D. and Granot, F. (1976), On solving fractional (0, 1) programs by implicit enumeration, *Canadian Journal of Operational Research and Information Processing 14*, 241-249.

[67] Granot, D. and Granot, F. (1977), On integer and mixed integer fractional programming problems, *Annals of Discrete Mathematics* 1, 221-231.

[68] Harant, J. (1998), A lower bound on the independence number of a graph, *Discrete Mathematics* 188, 239–243.

[69] Harant, J., Pruchnewski, A. and Voigt, M., (1999), On dominating sets and independent sets of graphs, Combinatorics, *Probability and Computing* 8, 547–553.

[70] Harant, J. (2000), Some news about the independence number of a graph, *Discussions Mathematical Graph Theory* 20, 71–79.

[71] Harary, F. and Ross, I.C. (1957), A procedure for clique detection using the group matrix, *Sociometry* 20, 205–215.

[72] Harte, L.J., Levine, R. and Kikta, R. (2001), *3G wireless Demystified*, McGraw-Hill Professional.

[73] Heinen, E. (1971), *Basics managerial decisions, the target system of company*, 2nd edition, Gabler, Wiesbaden.

[74] Hillier, F.S. and Lieberman, G.J. (2001), *Introduction to operations research*, Tata McGraw-Hill Education.

[75] Ho, L.T. and Claussen, H. (2007, September), Effects of user-deployed, co-channel femtocells on the call drop probability in a residential scenario. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on* 1-5.

[76] Hoskins, J.A. and Blom, R. (1984), Optimal allocation of warehouse personnel: A case study using fractional programming, *Focus* (UK) 3(2), 13–21.

[77] Houck, D.J. and Vemuganti, R.R. (1977), An algorithm for the vertex packing problem, *Operations Research* 25, 773-787.

[78] Ibaraki, T. (1983), Parametric approaches in fractional programs, *Mathematical Programming* 26(3), 345–362.

[79] Ipatov Valery, P. (2000), *Spread Spectrum and CDMA. Principles and Applications*, John Wiley & Sons, New York, USA.

[80] Isbell, J.R., and W.H. Marlow (1956), Attrition games, *Naval Research Logistics Quarterly* 3, 71–93.

[81] Kalvenes, J., Kennington, J. and Olinick, E. (2006), Base station location and service assignments in W-CDMA networks, *INFORMS Journal on Computing* 18(3), 366-376.

[82] Kikusts, P. (1986), Another algorithm determining the independence number of a graph, *Electronic information processing and cybernetics* 22, 157-166.

[83] Konno, H, and Inori, M. (1989), Bond portfolio optimization by bilinear fractional programming, *Journal of the Operations Research Society of Japan* 32(2), 143–158.

[84] Kornbluth, J.S.H. and Steuer, R.E. (1981), Multiple objective linear fractional programming, *Management Science* 27, 1024–1039.

[85] Komlosi, S., Rapcsak, T. and Schaible, S. (Eds.) (1994), Generalized convexity, *Proceedings of the 4th International Workshop at Pécs, Hungary, Lecture Notes Economics and Math Systems* 405, Springer, Berlin.

[86] Laguna, M. (1998), Applying robust optimization to capacity expansion of one location in telecommunications with demand uncertainty, *Management Science* 44(11-part-2), S101-S110.

[87] Li, H. (1994), A global approach for general 0-1 fractional programming, *European Journal of Operational Research* 73, 590–596.

[88] Loukakis, E. and Tsouros, C. (1982), Determining the number of internal stability of a graph, *International Journal of Computer Mathematics* 11, 207-220.

[89] Marcus, P.M. (1964), Derivation of maximal compatibles using Boolean algebra, *IBM Journal of Research and Development* 8, 537–538.

[90] Martos, B. (1975), *Nonlinear Programming: Theory and Methods*, North-Holland, Amsterdam.

[91] Mathar, R. and Schmeink, M. (2001). Optimal base station positioning and channel assignment for 3G mobile networks by integer programming, *Annals of Operations Research* 107(1-4), 225-236.

[92] Megiddo, N. (1979), Combinatorial optimization with rational objective functions, *Mathematics of Operations Research* 4, 414–424.

[93] Mjelde, K.M. (1983), *Methods of the Allocation of Limited Resources*, John Wiley & Sons, New York, USA.

[94] Motzkin, T.S. and Straus, E.G. (1965), Maxima for graphs and a new proof of a theorem of Turán, *Canadian Journal of Mathematics*, 17,533-540.

[95] Moon, J.W. and Moser, L. (1965), On cliques in graphs, *Israel Journal of Mathematics* 3, 23–28.

[96] Nemhauser, G.L. and Trotter, L.E. (1975), Vertex packing: structural properties and algorithms, *Mathematical Programming* 8, 232-248.

[97] Obaid, A. and Yanikomeroglu, H. (2000), Reverse-link power control in CDMA distributed antenna systems. *In Wireless Communications and Networking Confernce, 2000. WCNC. 2000 IEEE* 2, 608-612.

[98] Olinick, E.V. and Rosenberger, J.M. (2008), Optimizing revenue in CDMA networks under demand uncertainty, *European Journal of Operational Research* 186(2), 812-825.

[99] Pardalos, P.M. and Phillips, A.T. (1991), Global optimization of fractional programs, *Journal of Global Optimization* 1, 173–182.

[100] Pardalos, P.M., Bomze, I.M., Budinich, M. and Pelillo, M. (1999), The maximum clique problem, *Handbook of Combinational Optimization* 1-74.

[101] Pattillo, J. and Butenko, S. (2011), Maximum Clique, Maximum Independent Set, and Graph Coloring Problems, *Wiley Encyclopedia of Operations Research and Management Science*, John Wiley and Sons, New York, NY.

[102] Paull, M.C. and Unger, S.H. (1959), Minimizing the number of states in incompletely specified sequential switching functions, *IRE Transactions Electronic computers* EC-8, 356–367.

[103] Picard, J.C. and Queyranne, M. (1982), A network flow solution to some nonlinear 0−1 programming problems, with applications to graph theory, *Networks* 12(2), 141-159.

[104] Raciti, Robert C. (July 1995), *CELLULAR TECHNOLOGY*, Nova Southeastern University.

[105] Radzik, T. (1998), Fractional combinatorial optimization, In *Handbook Combinatorial Optimization* 429-478, Springer, US.

[106] Rao, M.R. (1971), Cluster analysis and mathematical programming, *Journal of the American Statistical Association* 66, 622–626.

[107] Rappaport, T.S. (1996), *Wireless communications: principles and practice* (Vol. 2). New Jersey: prentice hall PTR.

[108] Rebennack, S. (2009), Stable Set Problem: Branch & Cut Algorithms, In *Encyclopedia of Optimization* 3676-3688. Springer US.

[109] Riis, M. and Lodahl, J. (2002), A bicriteria stochastic programming model for capacity expansion in telecommunications, *Mathematical Methods of Operations Research* 56(1), 83-100.

[110] Robillard, P. (1971), (0, 1) hyperbolic programming problems, *Naval Research Logistics Quarterly*, 18(1), 47-57.

[111] Robson, J.M. (1986), Algorithms for maximum independent sets, *Journal of Algorithms* 7, 425–440.

[112] Robson, J.M. (2001), Finding a maximum independent set in time $o(2^{n/4})$. *Technical Report* 1251-01, LaBRI, University Bordeaux.

[113] Rossi, F. and Smriglio, S. (2001), A branch-and-cut algorithm for the maximum cardinality stable set problem, *Operations Research Letter* 28(2), 63-74.

[114] Roh, W. and Paulraj, A. (2002), MIMO channel capacity for the distributed antenna, In Vehicular Technology Conference, 2002. *Proceedings. VTC 2002-Fall. 2002 IEEE 56th* 2, 706-709.

[115] Rosenberger, J.M. and Olinick, E.V. (2007), Robust tower location for code division multiple access networks. *Naval Research Logistics* 54(2), 151-161.

[116] Rosenthal, R.E. (2007), *GAMS–A User's Guide*, GAMS Development Corporation, Washington, DC. Available online at the following website: http://www. gams. com.

[117] Rutherford, T.F. (1999), Applied general equilibrium modeling with MPSGE as a GAMS subsystem: An overview of the modeling framework and syntax, *Computational Economics* 14(1-2), 1-46.

[118] Sahinidis, N.V. (1996), BARON: A general purpose global optimization software package, *Journal of global optimization* 8(2), 201-205.

[119] Saleh, A.A., Rustako, A.J. and Roman, R. (1987), Distributed antennas for indoor radio communications, *IEEE Transactions on Communications*, 35(12), 1245-1251.

[120] Seidel, E. and Saad, E. (2010), LTE Home Node Bs and its enhancements in Release 9, *Nomor Research*.

[121] Schaible, S. (1976a), Fractional programming - I: *Duality, Management Science* 22(8), 858–867.

[122] Schaible, S. (1976b), Fractional programming – II: On Dinkelbach's algorithm, *Management Science* 22(8), 868–873.

[123] Schaible, S. (1978), Analysis and applications of quotient programs: A contribution to the planning with the help of nonlinear programming, *Mathematical Systems in Economics* 42, Anton Hain Verlag, Meisenheim.

[124] Schaible, S. (1981), Fractional programming: Applications and algorithms. *European Journal of Operational Research* 7, 111–120.

[125] Schaible, S. and Ziemba, W.T. (Eds.) (1981), Generalized concavity in optimization and economics, *Proceedings of NATO Advanced Study Institute in Vancouver*, Academy Press, New York.

[126] Schaible, S. (1983), Fractional programming, *Operations Research* 27(1): 39–54.

[127] Schaible, S. and Ibaraki, T. (1983), Fractional programming, *European Journal of Operational Research* 12(4), 325–338.

[128] Schaible, S. (1995), Fractional programming, In *Handbook of Global Optimization*, 495–608. Springer US.

[129] Schaible, S. (1996), Fractional programming with sums of ratios, In: *Castagnoli, E. and Giorgi, G. (Eds.), Scalar and Vector Optimization in Economic and Financial Problems, Proceedings of the Workshop in Milan, Italy*, 163–175.

[130] Schaible, S. and Frenk, H. (2009), Fractional Programming, *Encyclopedia of Optimization*: 1080-1091, Springer, Berlin.

[131] Scott, C.H., Jefferson, T.R. and Frenk, J.B.G. (1998), A duality theory for a class of generalized fractional programs, *Journal of Global Optimization* 12, 239–245.

[132] Selkow, S.M. (1994), A Probabilistic lower bound on the independence number of graphs, *Discrete Mathematics* 132, 363–365.

[133] Sen, S., Doverspike, R.D. and Cosares, S. (1994), Network planning with random demand, *Telecommunication Systems* 3(1), 11-30.

[134] Sherali, H. and Tuncbilek, C.H. (1992), A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique, *Journal of Global Optimization* 2, 101-112.

[135] Sherali, H. and Tuncbilek, C.H. (1997), New reformulation linearization convexification relaxations for univariate and multivariate polynomial programming problems, *Operations Research Letters* 21, 1-9.

[136] Sherali, H. (1998), Global optimization of nonconvex polynomial programming problems having rational exponents, *Journal of Global Optimization* 12, 267-283.

[137] Shor, N.Z. (1990), Dual quadratic estimates in polynomial and Boolean programming, *Annals of Operations Research* 25, 163-168.

[138] Shor, N.Z. (1998), *Nondifferentiable Optimization and Polynomial Problems*, Kluwer Academic Publishers, Dordrecht, The Netherlands.

[139] Singh, C. and Dass, B.K. (Eds.) (1989), *Continuous-time, fractional and multiobjective programming*, Analytic Publishing Company, Delhi.

[140] Sniedovich, M. (1988), Fractional

[141] programming revisited, *European Journal of Operational Research* 33(3), 334–341.

[142] Sohn, S. and Jo, G.S. (2006), Optimization of base stations positioning in mobile networks. *In Computational Science and Its Applications-ICCSA 2006* 779-787. Springer Berlin Heidelberg.

[143] Stancu-Minasian, I.M. (1997), *Fractional programming: Theory, Methods and Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands.

[144] St-Hilaire, M., Chamberland, S. and Pierre, S. (2006), Uplink UMTS network design-an integrated approach, *Computer Networks* 50(15), 2747-2761.

[145] Swarup, K. (1965), Linear Fractional Functional Programming, *Operations Research* 13(6), 1029-1036.

[146] Tawarmalani, M., Ahmed, S. and Sahinidis, N. (2002), Global Optimization of 0-1 Hyperbolic Programs, *Journal of Global Optimization* 24, 385–416.

[147] Tarjan, R.E. (1972), Finding a maximum clique, *Technical Report 72-123*, Computer Science Department, Cornell University, Ithaca, NY.

[148] Tarjan, R.E. and Trojanowski, A.E. (1977), Finding a maximum independent set, *SIAM Journal of Computing* 6, 537–546.

[149] Tomita, E., Tanaka, A. and Takahashi, H. (1988), The worst-time complexity for finding all the cliques, *Technical Report UEC-TR-C5, University of Electro-Communications*, Tokyo, Japan.

[150] Tomita, E., Tanaka, A. and Takahashi, H. (2006), The worst-case time complexity for generating all maximal cliques and computational experiments, *Theoretical Computer Science* 363(1), 28–42.

[151] Ugray, Z., Lasdon, L., Plummer, J.C., Glover, F., Kelly, J. and Marti, R. (2005), A multistart scatter search heuristic for smooth NLP and MINLP problems, In *Metaheuristic Optimization via Memory and Evolution* 25-57. Springer US.

[152] Von Neumann, J. (1937), About an economic system of equations and a generalization of Brouwer's fixed point theorem, In: *Menger K. (Eds.), Results of math colloquium* 8, 73-83.

[153] Wilf, H.S. (1986), *Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ.

[154] Wingo, D.R. (1985), Globally minimizing polynomials without evaluating derivatives, *International Journal of Computer Mathematics* 17, 287-294.

[155] Wu, T-H. (1997), A note on a global approach for general 0-1 fractional programming, *European Journal of Operational Research* 101(1), 220-223.

[156] Yanikomeroglu, H. and Sousa, E. S. (1993, October), CDMA distributed antenna system for indoor wireless communications. In *Universal Personal Communications, 1993. Personal Communications: Gateway to the 21st Century. Conference Record., 2nd International Conference on* 2, 990-994. IEEE.

[157] Yeh, S.P., Talwar, S., Lee, S.C. and Kim, H. (2008), WiMAX femtocells: a perspective on network architecture, capacity, and coverage, *Communications Magazine, IEEE* 46(10), 58-65.

[158] Zhang, Y., Hu, H. and Luo, J. (Eds.). (2007), *Distributed antenna systems: open architecture for future wireless communications*. CRC Press.

[159] Zhang, J. and Andrews, J.G. (2008), Distributed antenna systems with randomness. *IEEE Transactions on Wireless Communications*, 7(9), 3636-3646.

[160] Zhuang, H., Dai, L., Xiao, L. and Yao, Y. (2003), Spectral efficiency of distributed antenna system with random antenna layout, *Electronics Letters* 39(6), 495-496.