

Bi-Perspective Functions for Mixed-Integer Fractional Programs with Indicator Variables

Adam N. Letchford* Qiang Ni[†] Zhaoyu Zhong*

To appear in *Mathematical Programming Series A*

Abstract

Perspective functions have long been used to convert fractional programs into convex programs. More recently, they have been used to form tight relaxations of mixed-integer nonlinear programs with so-called *indicator variables*. Motivated by a practical application (maximising energy efficiency in an OFDMA system), we consider problems that have a fractional objective and indicator variables simultaneously. To obtain a tight relaxation of such problems, one must consider what we call a “bi-perspective” (Bi-P) function. An analysis of Bi-P functions leads to the derivation of a new kind of cutting planes, which we call “Bi-P-cuts”. Computational results indicate that Bi-P-cuts typically close a substantial proportion of the integrality gap.

Keywords: fractional programming, mixed-integer nonlinear programming, mobile wireless communications, OFDMA systems

1 Introduction

Let y be a vector of n continuous variables, and let $f(y)$ be a real function of y that is defined over a convex domain $C \subseteq \mathbb{R}_+^n$. The *perspective function* of f takes the form $tf(y/t)$, where t is a new continuous variable, and is defined over the domain $t \in \mathbb{R}_+$, $y \in tC$ [9, 17]. (By convention, the perspective function takes the value zero when $t = 0$ and y is the origin.) It is known that the perspective function of $f(y)$ is convex (or concave) if and only if $f(y)$ is convex (or concave) [17].

Perspective functions have a wide range of uses in, e.g., convex analysis, optimisation, statistics, signal processing and machine learning (see [4] for a recent survey). In this paper, we focus on two uses in optimisation. The first is to convert *fractional programs* into convex optimisation problems,

*Department of Management Science, Lancaster University, Lancaster LA1 4YX. E-mail: {A.N.Letchford,z.zhong1}@lancaster.ac.uk

[†]School of Computing and Communications, Lancaster University, Lancaster LA1 4WA. E-mail: q.ni@lancaster.ac.uk

and thereby render them easier to solve [3, 19]. The second is to reformulate certain *mixed-integer nonlinear programs* (MINLPs), in such a way that the continuous relaxation is strengthened [6, 7]. The MINLPs in question are those with so-called *indicator variables*.

Recently, while studying certain problems arising in the context of mobile wireless communications, we encountered an MINLP that exhibited both of these features (i.e., a fractional objective and indicator variables) simultaneously. It turns out that, in order to obtain tight convex relaxations of such problems, one needs to study a new kind of function, which we call a *bi-perspective* (Bi-P) function. These functions are the topic of this paper.

Unfortunately, it turns out that the Bi-P function of a concave function is not concave in general. To deal with this, we characterise the concave envelope of a Bi-P function over a rectangular domain. We then derive a family of linear inequalities, which we call *Bi-P cuts*, that completely describe the concave envelope. We also show how to generalise the Bi-P cuts when there are “multiple-choice” constraints, stating that two or more indicator variables cannot take the value 1 simultaneously. Finally, we report the results of some computational experiments. It turns out that, for our particular problem, the new cuts typically close over 95% of the integrality gap.

The paper is structured as follows. The relevant literature is reviewed in Section 2. In Section 3, we present the theoretical results concerning Bi-P functions and Bi-P cuts. In Section 4, we consider the multiple-choice case. In Section 5, we give details of our practical application. Computational results are given in Section 6, and concluding remarks are made in Section 7.

2 Literature Review

Now we briefly review the relevant literature. We cover perspective functions and perspective cuts in Subsections 2.1 and 2.2, respectively. In Subsection 2.3, we give some background on optimisation in mobile wireless communications.

2.1 Perspective functions

As mentioned in the introduction, two particular uses of perspective functions will turn out to be of importance to us. These are as follows.

1. Consider a *fractional program* of the form:

$$\max \left\{ f(y)/g(y) : x \in C \right\},$$

where $C \subseteq \mathbb{R}^n$ is convex, $f(y)$ is non-negative and concave over the domain C , and $g(y)$ is positive and convex over the domain C . It is

known [3, 19] that such a problem can be reformulated as

$$\max \left\{ tf(y'/t) : tg(y'/t) \leq 1, y' \in tC, t > 0 \right\},$$

where t is a new non-negative variable representing $1/g(y)$, and y' is a new vector of variables representing $y/g(y)$. The reformulated problem can often be solved efficiently, since the objective function is concave and the feasible region is convex.

2. Consider an MINLP of minimisation type, in which the cost function contains a term $f(y)$, where y is a vector of continuous variables and f is a convex function. Suppose that the MINLP also contains an *indicator variable*, i.e., a binary variable x with the property that, if x takes the value zero, then all of the components of y must also take the value zero. Then the continuous relaxation of the MINLP is strengthened, while maintaining convexity, if we replace $f(y)$ with the perspective function $xf(y/x)$ [6, 7]. (For a generalisation, see [1].)

2.2 Perspective cuts

One problem with perspective functions is that they are non-differentiable at the origin. Moreover, they become increasingly ill-conditioned as t approaches zero from above. This can cause algorithms for convex optimisation (or indeed convex MINLP) to run into serious numerical difficulties. To get around this, Frangioni & Gentile [6] proposed to approximate perspective functions using linear inequalities. They show that imposing a non-linear constraint of the form $z \geq xf(y/x)$, where f is a convex function and x is an indicator variable, is equivalent to imposing the linear constraints

$$z \geq \nabla f(\bar{y}) \cdot y + \left(f(\bar{y}) - \nabla f(\bar{y}) \cdot \bar{y} \right) x \quad (1)$$

for all \bar{y} in the domain of y . The constraints (1) are called *perspective cuts*. Although the perspective cuts are infinite in number, they can be very useful as cutting planes within an exact algorithm for convex MINLPs with indicator variables (see again [6, 7]).

Note that the classical *Kelley cuts* [10] for the function $f(y)$ take the form

$$z \geq f(\bar{y}) + \nabla f(\bar{y}) \cdot (y - \bar{y}).$$

Thus, the perspective cuts can be viewed as strengthened Kelley cuts.

2.3 Optimisation in mobile wireless communications

In *mobile wireless communications*, mobile devices (such as smartphones or tablets) communicate with one another via so-called *base stations*. Each base station must periodically allocate its available resources (time, power

and bandwidth) in order to receive and transmit data efficiently (see, e.g., [5, 8, 16]).

These days, many base stations follow an *Orthogonal Frequency-Division Multiple Access* (OFDMA) architecture. In an OFDMA system, we have a set I of communication channels, called *subcarriers*, and a set J of *users*. Each subcarrier can be assigned to at most one user, but a user may be assigned to more than one subcarrier. If we let p_i denote the power (in watts) assigned to subcarrier i , the classical *Shannon–Hartley theorem* [21] states that the maximum data rate (in bits per second) that can be transmitted from subcarrier i is:

$$f_i(p_i) = B_i \log_2(1 + p_i/N_i),$$

where B_i is the bandwidth of subcarrier i (in hertz), and N_i is the level of noise in channel i (in watts). We remark that $f_i(p_i)$ is concave over the domain $p_i \geq 0$. Moreover, if we let $S_j \subset I$ denote the set of subcarriers allocated to user j , the total data rate for user j is just $\sum_{i \in S_j} f_i(p_i)$.

A wide variety of optimisation problems concerned with OFDMA systems have been considered, with various objectives and side-constraints (see, e.g., [11, 12, 14, 15, 20, 22–26]). Recently, driven by environmental considerations, some authors working on OFDMA systems have focused on maximising *energy efficiency*, which is defined as total data rate divided by total power (e.g., [22–24, 26]). This leads immediately to a fractional objective function, which is what led us to the present paper.

3 Bi-Perspective Functions and Cuts

This section is concerned with bi-perspective (Bi-P) functions and cuts. In Subsection 3.1, we define Bi-P functions and point out that they are neither convex nor concave in general. In Subsection 3.2, we show how to compute concave over-estimators of Bi-P functions. Then, in Subsection 3.3, we present the Bi-P cuts.

3.1 Bi-P functions

To begin, we give a formal definition of Bi-P functions.

Definition 1 *Let y be a vector of n continuous variables, let $f(y)$ be a real function of y that is defined over a convex domain $C \subseteq \mathbb{R}_+^n$, let x be an indicator variable, and let t be a continuous variable with domain $[\ell, u]$, where $0 < \ell < u$. The function*

$$g(x, t, y) = x t f\left(\frac{y}{x t}\right),$$

with domain $x \in [0, 1]$, $t \in [\ell, u]$ and $y \in xtC$, will be called the “bi-perspective” (Bi-P) function of $f(y)$. (By convention, the Bi-P function takes the value zero when $xt = 0$ and y is the origin.)

Whereas standard perspective functions preserve convexity and/or concavity, the same is not true for Bi-P functions. This is shown by the following example.

Example 1 Let $f(y) = 1$ for all $y \in C$. The Bi-P function $g(x, t, y)$ is just xt , with domain $x \in [0, 1]$, $t \in [\ell, u]$ and $y \in xtC$. Since it is an indefinite quadratic function, it is neither convex nor concave over the given domain.

3.2 Concave envelope

Now suppose that $f(y)$ is concave over C . Since the corresponding Bi-P function $g(x, t, y)$ is not guaranteed to be concave over its domain, it is natural to seek the strongest possible concave over-estimating function (sometimes called the *concave envelope*). We will let $h(x, t, y)$ denote this function. The following theorem expresses $h(x, t, y)$ as the solution to an optimisation problem in the single continuous variable t' .

Theorem 1 For any concave function $f(y)$ with convex domain C , the concave envelope $h(x, t, y)$ of the Bi-P function of f is equal to:

$$\max \quad x t' f\left(\frac{y}{x t'}\right) \quad (2)$$

$$\text{s.t.} \quad \frac{t-u(1-x)}{x} \leq t' \leq \frac{t-\ell(1-x)}{x} \quad (3)$$

$$t' \in [\ell, u]. \quad (4)$$

Proof. Theorem 1 of [2], together with the disjunction $(x = 0) \vee (x = 1)$, implies that $h(x, t, y)$ is equal to:

$$\begin{aligned} \max \quad & \theta_0 g(x_0, t_0, y^0) + \theta_1 g(x_1, t_1, y^1) \\ \text{s.t.} \quad & \theta_0 + \theta_1 = 1 \\ & \theta_0 x_0 + \theta_1 x_1 = x \\ & \theta_0 t_0 + \theta_1 t_1 = t \\ & \theta_0 y^0 + \theta_1 y^1 = y \\ & x_k = k \quad (k = 0, 1) \\ & t_k \in [\ell, u] \quad (k = 0, 1) \\ & y^k \in x_k t_k C \quad (k = 0, 1). \end{aligned}$$

Eliminating x_0 and x_1 , and using the fact that y^0 must be the origin, this reduces to:

$$\max \quad \theta_1 t_1 f(y^1/t_1)$$

$$\begin{aligned}
\text{s.t.} \quad & \theta_0 + \theta_1 = 1 \\
& \theta_1 = x \\
& \theta_0 t_0 + \theta_1 t_1 = t \\
& \theta_1 y^1 = y \\
& t_k \in [\ell, u] \quad (k = 0, 1) \\
& y^1 \in t_1 C.
\end{aligned}$$

We can now replace θ_1 and θ_0 with x and $1 - x$, respectively, to give:

$$\begin{aligned}
\max \quad & x t_1 f(y^1/t_1) \\
\text{s.t.} \quad & (1 - x)t_0 + x t_1 = t \\
& x y^1 = y \\
& t_k \in [\ell, u] \quad (k = 0, 1) \\
& y^1 \in t_1 C.
\end{aligned}$$

Next, we can replace y^1 with y/x , to give:

$$\begin{aligned}
\max \quad & x t_1 f\left(\frac{y}{x t_1}\right) \\
\text{s.t.} \quad & (1 - x)t_0 + x t_1 = t \\
& t_k \in [\ell, u] \quad (k = 0, 1).
\end{aligned}$$

Finally, we eliminate t_0 , and replace t_1 with t' to get the result. \square

In general, one must use calculus to solve the optimisation problem (2)–(4). In some important special cases, however, it is possible to obtain a closed-form solution. To show this, we will need the following two definitions.

Definition 2 *The function $f(y)$ is said to be t -increasing if*

$$(t + \epsilon) f\left(\frac{y}{t + \epsilon}\right) \geq t f(y/t)$$

holds for all $t \in [\ell, u]$, $0 \leq \epsilon \leq u - t$ and $y \in tC \cap (t + \epsilon)C$.

Definition 3 *The function $f(y)$ is said to be t -decreasing if*

$$(t + \epsilon) f\left(\frac{y}{t + \epsilon}\right) \leq t f(y/t)$$

holds for all $t \in [\ell, u]$, $0 \leq \epsilon \leq u - t$ and $y \in tC \cap (t + \epsilon)C$.

We then have the following two results:

Proposition 1 *If $f(y)$ is concave and t -increasing over convex domain C , then*

$$h(x, t, y) = \min \left\{ u x f\left(\frac{y}{u x}\right), (t - \ell(1 - x)) f\left(\frac{y}{t - \ell(1 - x)}\right) \right\}. \quad (5)$$

Proof. In this case, the optimum in (2)–(4) is obtained by setting t' to its maximum possible value, which is $\min \left\{ u, (t - \ell(1 - x))/x \right\}$. \square

Proposition 2 *If $f(y)$ is concave and t -decreasing over convex domain C , then*

$$h(x, t, y) = \min \left\{ \ell x f \left(\frac{y}{\ell x} \right), (t - u(1 - x)) f \left(\frac{y}{t - u(1 - x)} \right) \right\}.$$

Proof. In this case, the optimum in (2)–(4) is obtained by setting t' to its minimum possible value, which is $\max \left\{ \ell, (t - u(1 - x))/x \right\}$. \square

An obvious sufficient condition for a concave function to be t -increasing is for it to be non-negative and non-decreasing. The following proposition gives a rather different sufficient condition.

Proposition 3 *If $0 \in C$ and $f(0) \geq 0$, then $f(y)$ is t -increasing.*

Proof. Since $f(y)$ is concave, we have

$$f \left(\frac{y}{t + \epsilon} \right) \geq \frac{\epsilon}{t + \epsilon} f(0) + \frac{t}{t + \epsilon} f \left(\frac{y}{t} \right).$$

Given that $\epsilon > 0$, $t \geq \ell > 0$ and $f(0) \geq 0$, we have

$$f \left(\frac{y}{t + \epsilon} \right) \geq \frac{t}{t + \epsilon} f(y/t).$$

Multiplying both sides by $t + \epsilon$ yields the stated result. \square

Checking whether $f(y)$ is t -decreasing, on the other hand, is more difficult. In particular, it is not sufficient for $f(y)$ to be non-positive and non-decreasing. For example, the function $f(y) = -2^y$ is non-positive and non-decreasing over the domain \mathbb{R}_+ , but it is not t -decreasing, since $0.6f(1/0.6) \approx -1.905 > 0.5f(1/0.5) = -2$. The same function also shows that it is not sufficient to have $0 \in C$ and $f(0) \leq 0$. On the other hand, t -decreasing functions do exist. A trivial example is $f(y) = -1$. A more complex example is $f(y) = -\sqrt{y^2 + 1}$.

3.3 Bi-P cuts

Observe that the nonlinear function (5) is non-differentiable not only when $xt = 0$, but also when $u = t - \ell(1 - x)$. This suggests that standard NLP solvers could struggle to handle functions of the form (5). Moreover, there are situations in which one might prefer to use an LP (or MILP) solver rather

than an NLP (or MINLP) solver. So, following Frangioni & Gentile [6], we consider the *hypograph* of $h(x, t, y)$, i.e., the set

$$H = \{(x, t, y, z) : x \in [0, 1], t \in [\ell, u], y \in x t C, z \leq h(x, t, y)\}.$$

The following proposition gives a complete description of H by linear inequalities, for the case in which $f(y)$ is t -increasing.

Proposition 4 *When $f(y)$ is t -increasing over C , the hypograph H is described by the linear inequalities*

$$z \leq \nabla f(\bar{y}) y + \left(f(\bar{y}) - \nabla f(\bar{y}) \bar{y} \right) u x \quad (6)$$

$$z \leq \nabla f(\bar{y}) y + \left(f(\bar{y}) - \nabla f(\bar{y}) \bar{y} \right) \left(t - \ell(1 - x) \right). \quad (7)$$

for $\bar{y} \in C$.

Proof. Let H^+ denote the set of 6-tuples $(x, t, y, z, \alpha_1, \alpha_2)$ satisfying

$$x \in [0, 1], t \in [\ell, u], y \in x t C \quad (8)$$

$$\alpha_1 = u x \quad (9)$$

$$\alpha_2 = t - \ell(1 - x) \quad (10)$$

$$z \leq \alpha_k f(y/\alpha_k) \quad (k = 1, 2). \quad (11)$$

Given that the equations (9) and (10) are linear, H^+ is an affine image of H . Moreover, using exactly the same argument as in [6], one can show that H^+ is described by the constraints (8) together with the following perspective cuts:

$$z \leq \nabla f(\bar{y}) y + \left(f(\bar{y}) - \nabla f(\bar{y}) \bar{y} \right) \alpha_k \quad (k = 1, 2; \bar{y} \in C).$$

Eliminating α_1 and α_2 yields the result. \square

We will call constraints (6) and (7) *type-1* and *type-2* Bi-P cuts, respectively. Note that the type-1 cuts can be derived as standard perspective cuts from the modified perspective function $u x f(y/(u x))$. The type-2 cuts, on the other hand, are harder to interpret.

We now make some remarks about Bi-P cuts.

Remark 1 *The type 1 Bi-P cuts pass through all points satisfying $t = \ell$, $y = 0$ and $z = 0$, along with the point $x = 1$, $t = u$, $y = u \bar{y}$ and $z = u f(\frac{\bar{y}}{u})$. Thus, they define maximal faces of H whenever the point $(\bar{y}, f(\bar{y}))$ lies on a maximal face of the hypograph of the original function $f(y)$. A similar argument applies to the type-2 Bi-P cuts.*

Remark 2 If we reduce the domain of t , the Bi-P cuts become stronger. Thus, if one wishes to make the cuts as tight as possible, it may be worthwhile applying “domain reduction” techniques (see, e.g., [18]) to t .

Remark 3 Let (x^*, t^*, y^*, z^*) be the solution to some relaxation. To solve the separation problem for type-1 Bi-P cuts, it suffices to compute the quantity $ux^* f\left(\frac{y^*}{ux^*}\right)$. If this quantity is less than z^* , then a cut is violated and the vector \bar{y} yielding the cut is y^*/ux^* . Similarly, to solve the separation problem for type-2 Bi-P cuts, it suffices to compute $(t^* - u(1 - x^*)) f\left(\frac{y^*}{t^* - u(1 - x^*)}\right)$ and compare it with z^* .

4 Bi-P cuts and Multiple-Choice Constraints

It is very common in integer programming to encounter constraints which state that at most one of a set of binary variables can take a positive value. Such constraints go by various names, such as *multiple-choice* constraints, *clique* constraints or *generalised upper bounds*. In this subsection, we consider the case in which each of the variables in the given set is an indicator variable.

To be more precise, suppose we have:

- positive constants ℓ, u with $\ell < u$;
- an integer $m \geq 2$ and positive integers n_1, \dots, n_m ;
- a convex domain $C_j \subseteq \mathbb{R}^{n_j}$ for $j = 1, \dots, m$;
- a t -increasing concave function $f_j : \mathbb{R}^{n_j} \rightarrow \mathbb{R}$ for $j = 1, \dots, m$.

Let Q denote the set of quadruples (x, y, z, t) satisfying

$$t \in [\ell, u] \tag{12}$$

$$\sum_{j=1}^m x_j \leq 1 \tag{13}$$

$$z_j \leq t f_j(y^j/t) \quad (j = 1, \dots, m) \tag{14}$$

$$x_j \in \{0, 1\} \quad (j = 1, \dots, m) \tag{15}$$

$$y^j \in t C_j \quad (j = 1, \dots, m) \tag{16}$$

$$z_j \in \mathbb{R} \quad (j = 1, \dots, m) \tag{17}$$

$$x_j = 0 \Rightarrow z_j = 0 \quad (j = 1, \dots, m)$$

$$x_j = 0 \Rightarrow y_i^j = 0 \quad (j = 1, \dots, m; i = 1, \dots, n_j).$$

Note that all points in Q satisfy the (non-convex) constraints

$$\begin{aligned} y^j &\in t x_j C_j & (j = 1, \dots, m) \\ z_j &\leq t x_j f_j\left(\frac{y^j}{t x_j}\right) & (j = 1, \dots, m), \end{aligned}$$

provided that, as usual, we use the convention that the right-hand side evaluates to zero when tx_j is zero. Together with Remark 3 in Subsection 3.2, this implies that all points in Q satisfy the convex constraints:

$$z_j \leq ux_j f_j \left(\frac{y^j}{ux_j} \right) \quad (j = 1, \dots, m) \quad (18)$$

$$z_j \leq (t - \ell(1 - x_j)) f_j \left(\frac{y^j}{t - \ell(1 - x_j)} \right) \quad (j = 1, \dots, m). \quad (19)$$

From this, one can derive one family of type-1 and type-2 Bi-P cuts for each value of j .

Perhaps surprisingly, the addition of the constraints (18), (19) to the system (12)–(17) does not yield a complete description of the convex hull of Q . This is shown in the following example.

Example: Let ϵ be a small positive constant. Suppose that (i) $\ell = \epsilon$ and $u = 1$; and (ii) $n_j = 1$, $C_j = [0, 1]$ and $f_j(y_1^j) = (y_1^j)^\epsilon$ for all j . Consider the fractional point obtained by setting t to ϵ , all x and y variables to $1/m$, and all z variables to $\epsilon/(m\epsilon)^\epsilon$. One can check that this point satisfies (18) and (19) for all j , and therefore all Bi-P cuts. Now, observe that all points in Q satisfy the following convex inequality:

$$\sum_{j=1}^m z_j \leq t \left(\frac{\sum_{j=1}^m y_1^j}{t} \right)^\epsilon.$$

The above-mentioned fractional point does not satisfy this, since the left- and right-hand sides evaluate to $(m\epsilon)^{1-\epsilon}$ and $\epsilon^{1-\epsilon}$, respectively. Accordingly, the point cannot lie in the convex hull of Q . \square

Now, for any $j \in \{1, \dots, m\}$ and any $\bar{y}^j \in C^j$, let $\|\bar{y}^j\|$ denote $f_j(\bar{y}^j) - \nabla f_j(\bar{y}^j) \cdot \bar{y}^j$. The following theorem presents a huge family of valid inequalities, which generalise the type-2 Bi-P cuts (7).

Theorem 2 *Let S be any non-empty subset of $\{1, \dots, m\}$. For each $j \in S$, let \bar{y}^j be any point in C^j such that $\|\bar{y}^j\| > 0$. Then the linear inequality*

$$\sum_{j \in S} \frac{z_j}{\|\bar{y}^j\|} \leq \sum_{j \in S} \frac{\nabla f_j(\bar{y}^j) \cdot y^j}{\|\bar{y}^j\|} + t - \ell \left(1 - \sum_{j \in S} x_j \right) \quad (20)$$

is satisfied by all points in Q .

Proof. We consider two cases:

Case 1: $x_j = 0$ for all $j \in S$. This forces y_i^j to be zero for all $j \in S$ and for $i = 1, \dots, n_j$. This in turn forces z_j to be zero for all $j \in S$. Thus, the

inequality reduces to $t \geq a$, which is trivially valid.

Case 2: $x_j = 1$ for some $j \in S$. This forces x_k to be zero for all $k \in S \setminus \{j\}$, along with the associated y and z variables. Thus, the inequality reduces to

$$\frac{z_j}{\|\bar{y}^j\|} \leq \frac{\nabla f_j(\bar{y}^j) \cdot y^j}{\|\bar{y}^j\|} + t.$$

Multiplying this by $\|\bar{y}^j\|$ we obtain:

$$z_j \leq \nabla f_j(\bar{y}^j) \cdot y^j + \|\bar{y}^j\| t.$$

This last inequality can be derived from the (convex) constraint $z_j \leq t f_j(y^j/t)$, in exactly the same way as the perspective cuts. \square

We call the constraints (20) *multiple-choice* Bi-P cuts, or *MC* cuts for short. They reduce to type-2 Bi-P cuts when $|S| = 1$.

The separation problem for MC cuts can be solved in polynomial time as follows. Let (x^*, y^*, z^*, t^*) be the point to be separated. For $j = 1, \dots, m$, compute $\ell x_j^* f((y^j)^*/(\ell x_j^*))$. If this quantity is less than z_j^* , then insert j into S and set \bar{y}^j to $(y^j)^*/(\ell x_j^*)$; otherwise, do not insert j into S . Once this has been done for all j , check the MC cut for violation.

5 Application to OFDMA Systems

We now apply the theoretical results in the last section to an optimisation problem associated with OFDMA systems. In Subsection 5.1, we define our problem formally and model it as a mixed 0-1 fractional program with indicator variables. In Subsection 5.2, we reformulate the problem as a semi-infinite mixed 0-1 linear program. In Subsection 5.3, we show how to strengthen the semi-infinite formulation using Bi-P cuts.

5.1 The problem

Let I, J, B_i, N_i and f_i be defined as in Subsection 2.3. Let $P > 0$ denote the maximum power available, and let $\sigma \in (0, P)$ denote the *system power*, which is the amount of power needed by the OFDMA system regardless of actual data rates. Finally, suppose that each user $j \in J$ has a non-negative *demand* d_j . The task is to maximise the energy efficiency, subject to the constraint that the total data rate for each user j is at least d_j .

We call this problem the *fractional subcarrier and power allocation problem with rate constraints* (F-SPARC). (A related problem, called the SPARC, was studied in [12]. The difference is that the objective in the SPARC was simply to maximise the total data rate.)

A natural formulation of the F-SPARC is obtained as follows. For all $i \in I$ and $j \in J$, let x_{ij} be a binary variable, indicating whether user j is

assigned to subcarrier i . Also let p_{ij} be a non-negative continuous variable, which represents the amount of power supplied to subcarrier i (if $x_{ij} = 1$), or zero otherwise. We then have:

$$\max \quad \frac{\sum_{i \in I} \sum_{j \in J} f_i(p_{ij})}{\sigma + \sum_{i \in I} \sum_{j \in J} p_{ij}} \quad (21)$$

$$\text{s.t.} \quad \sum_{i \in I} \sum_{j \in J} p_{ij} \leq P - \sigma \quad (22)$$

$$\sum_{j \in J} x_{ij} \leq 1 \quad (\forall i \in I) \quad (23)$$

$$\sum_{i \in I} f_i(p_{ij}) \geq d_j \quad (\forall j \in J) \quad (24)$$

$$p_{ij} \leq P x_{ij} \quad (\forall i \in I, j \in J) \quad (25)$$

$$p_{ij} \in \mathbb{R}_+ \quad (\forall i \in I, j \in J) \quad (26)$$

$$x_{ij} \in \{0, 1\} \quad (\forall i \in I, j \in J). \quad (27)$$

The objective function (21) represents the total data rate divided by the total power used (including system power). The constraint (22) ensures that the total power used does not exceed the amount available. Constraints (23) ensure that each subcarrier is assigned to at most one user. Constraints (24) ensure that the user demands are met. Constraints (25)–(27) are self-explanatory.

The problem (21)–(27) is a *mixed 0-1 fractional program*. The x variables are clearly *indicator variables*, since setting any x variable to zero forces the corresponding p variable to zero. Moreover, the constraints (23) are clearly *multiple-choice* constraints.

From now on, we let $D = \sum_{j \in J} d_j$ denote the total user demand. Note that an upper bound for the F-SPARC can be computed by solving the following (continuous) fractional program:

$$\max \left\{ \frac{\sum_{i \in I} f_i(p_i)}{\sigma + \sum_{i \in I} p_i} : \sum_{i \in I} p_i \leq P - \sigma, \sum_{i \in I} f_i(p_i) \geq D, p \in \mathbb{R}_+^{|I|} \right\}.$$

This fractional program can be converted into a convex program using the transformation mentioned in Subsection 2.1. We denote the corresponding upper bound by U .

5.2 Reformulation

We now reformulate the problem to make it easier to solve. This is done in three steps.

The first step is to convert the fractional objective function into a concave function. To do this, we use the transformation mentioned in Subsection 2.1. Let t be a non-negative continuous variable, representing $1/(\sigma + \sum_{i \in I} \sum_{j \in J} p_{ij})$, i.e., the reciprocal of the total power used. Also, for all i and j , let \tilde{p}_{ij} be a non-negative continuous variable, representing $t p_{ij}$. The

problem is then equivalent to the following:

$$\max \quad \sum_{i \in I} \sum_{j \in J} t f_i(\tilde{p}_{ij}/t) \quad (28)$$

$$\text{s.t.} \quad (23), (27)$$

$$\sigma t + \sum_{i \in I} \sum_{j \in J} \tilde{p}_{ij} = 1 \quad (29)$$

$$1/P \leq t \leq 1/\sigma \quad (30)$$

$$\sum_{i \in I} f_i(\tilde{p}_{ij}/t) \geq d_j \quad (\forall j \in J) \quad (31)$$

$$0 \leq \tilde{p}_{ij} \leq x_{ij} \quad (\forall i \in I, j \in J). \quad (32)$$

The objective function (28) is now concave, and all constraints are linear, apart from (31), which are convex.

The second step is the following. For $i \in I$ and $j \in J$, define a new variable, say z_{ij} , representing the quantity $t f_i(\tilde{p}_{ij}/t)$. The problem then becomes:

$$\max \quad \sum_{i \in I} \sum_{j \in J} z_{ij}$$

$$\text{s.t.} \quad (23), (27), (29), (30), (32)$$

$$\sum_{i \in I} z_{ij} \geq d_j t \quad (\forall j \in J)$$

$$z_{ij} \leq t f_i(\tilde{p}_{ij}/t) \quad (\forall i \in I, j \in J) \quad (33)$$

$$z_{ij} \in \mathbb{R}_+ \quad (\forall i \in I, j \in J).$$

Now, all of the nonlinearity has been “concentrated” in the (convex) constraints (33).

Finally, we replace the constraints (33) with the following linear constraints:

$$z_{ij} \leq f'_i(\bar{p}) \tilde{p}_{ij} + (f_i(\bar{p}) - f'_i(\bar{p}) \bar{p}) t \quad (\forall i \in I, j \in J, \bar{p} \in [0, P - \sigma]). \quad (34)$$

These linear constraints can be derived in the same way as standard perspective cuts. Since they are infinite in number, we have formulated the F-SPARC as a *semi-infinite mixed 0-1 linear program*. It can be solved (to arbitrary fixed accuracy) with an LP-based branch-and-cut algorithm.

5.3 Bi-P cuts

We now use the results in Section 3 to strengthen our semi-infinite formulation of the F-SPARC.

We start by reducing the domain of t . Recall that D denotes the total user demand. We compute the following lower bound on the total amount of power used:

$$P_{\min} = \sigma + \min \left\{ \sum_{i \in I} p_i : \sum_{i \in I} f_i(p_i) \geq D, p_i \geq 0 (i \in I) \right\}.$$

The upper bound on t in (30) can then be reduced from $1/\sigma$ to $1/P_{\min}$.

Now, let i and j be fixed, and consider the constraints (27), (32) and (33). It is clear that all feasible triples $(x_{ij}, \tilde{p}_{ij}, r_{ij})$ satisfy

$$z_{ij} \leq t x_{ij} f_i \left(\frac{\tilde{p}_{ij}}{t x_{ij}} \right),$$

where, as usual, the convention is that the function on the right-hand side takes the value zero when $\tilde{p}_{ij} = t x_{ij} = 0$. The function is clearly a t -increasing Bi-P function. The only tricky part is that the natural domain of \tilde{p}_{ij} is $[0, 1]$, which is not equal to $t x_{ij} C$ for some convex domain C . Despite this complication, we have the following result:

Proposition 5 *For all $i \in I$, $j \in J$ and $\bar{p} \in [0, P - \sigma]$, the following type-1 and type-2 Bi-P cuts are valid for the F-SPARC:*

$$z_{ij} \leq f'_i(\bar{p}) \tilde{p}_{ij} + \left(\frac{f_i(\bar{p}) - f'_i(\bar{p}) \bar{p}}{P_{\min}} \right) x_{ij} \quad (35)$$

$$z_{ij} \leq f'_i(\bar{p}) \tilde{p}_{ij} + (f_i(\bar{p}) - f'_i(\bar{p}) \bar{p}) \left(t - \frac{1 - x_{ij}}{P} \right). \quad (36)$$

Proof. We already know that t must lie in the interval $[1/P, 1/P_{\min}]$. Also, $p_{ij} \leq P - \sigma$, or, equivalently, $\tilde{p}_{ij} \in t[0, P - \sigma]$. In fact, given that \tilde{p}_{ij} must be zero when x_{ij} is zero, we can conclude that $\tilde{p}_{ij} \in t x_{ij} [0, P - \sigma]$. We can now generate Bi-P cuts as usual. \square

Note that the type-2 cuts (36) dominate (34).

Finally, since the constraints (23) are multiple-choice constraints, we can also generate MC cuts. In our preliminary experiments, we found that the most useful MC cuts were those with (a) $S = J$ and (b) \bar{p}_{ij} equal to the same (positive) value for all $j \in J$. These cuts take the form:

$$\sum_{j \in J} z_{ij} \leq f'_i(\bar{p}) \sum_{j \in J} \tilde{p}_{ij} + \left(f_i(\bar{p}) - f'_i(\bar{p}) \bar{p} \right) t \quad (i \in I, \bar{p} \in [0, P - \sigma]). \quad (37)$$

We remark that the cuts (37) collectively enforce the following convex inequalities:

$$\sum_{j \in J} z_{ij} \leq t f_i \left(\frac{\sum_{j \in J} \tilde{p}_{ij}}{t} \right) \quad (i \in I). \quad (38)$$

From this it can be shown that the upper bound obtained by adding all of the cuts (37) to the continuous relaxation of the semi-infinite formulation is equal to U , the upper bound mentioned at the end of Subsection 5.1.

We remark that the separation problem for the cuts (37) can be solved simply by setting \bar{p} to the current value of $\sum_{j \in J} \tilde{p}_{ij}$, and checking the inequality (38) for violation. This can be done in $O(|J|)$ time for a given i .

6 Computational Experiments

In the previous section, we described three families of cutting planes for the F-SPARC: the type-1 cuts (35), the type-2 cuts (36), and the special MC cuts (37). In this section, we present some computational results to shed light on the relative usefulness of these different kinds of cuts.

6.1 Test instances

To construct our test instances, we used the procedure described in [12], which is designed to produce instances typical of a small base station. In detail, the instances have $|I| = 72$, $|J| \in \{4, 6\}$, and P set to 36 watts. The noise powers N_i are random numbers distributed uniformly in $(10^{-6}, 10^{-5})$, and the bandwidths B_i are all set to 1.25MHz. The user demands follow a lognormal distribution.

Let D be the total demand, as before, and let M be the maximum possible data rate of the system. As in [12], we call the quantity D/M the *demand ratio* (DR) of the given instance. The user demands are scaled so that the DR takes values in $\{0.75, 0.8, 0.85, 0.9, 0.95\}$. (The closer the DR is to 1, the harder the instance tends to be.) For each combination of $|J|$ and DR, we constructed 10 random instances. This makes $2 \times 5 \times 10 = 100$ instances in total. These instances have been made available at:

<http://www.research.lancs.ac.uk/portal/en/datasets/search.html>

under “OFDMA Optimisation”.

6.2 Experimental setup

For each instance, we did the following. We began by computing the upper bound U mentioned at the end of Subsection 5.1, using MOSEK. We then ran the heuristic described in [13] to compute a lower bound. If the bounds differed by more than 0.1%, we ran an exact algorithm, similar to the one described in [12], until the difference between the lower and upper bounds dropped below 0.1%. We remark that this exact algorithm took a long time (sometimes several minutes) to converge for some of our test instances.

The next step was to solve, for each instance, the continuous relaxation of the formulation described in Subsection 5.2. To do this, we used a (fairly standard) LP-based cutting-plane algorithm, in which the inequalities (34) were added dynamically as cutting planes. In each major iteration, all inequalities violated by more than 10^{-4} were added to the LP. A time limit of two minutes per instance was also imposed. (In most cases, tailing off occurred well before the time limit.)

The cutting-plane algorithm was coded in Julia v0.5 and run on a virtual machine cluster with 16 CPUs (ranging from Sandy Bridge to Haswell

Table 1: Average percentage gaps when $|J| = 4$

Cuts	Demand Ratio				
	0.75	0.80	0.85	0.90	0.95
\emptyset	165	165	168	172	175
T1	0.00	0.00	0.09	0.03	0.02
T2	0.01	0.01	0.03	0.01	0.03
MC	2.19	1.39	1.06	0.25	0.06
T1, T2	0.00	0.00	0.03	0.01	0.01
All	0.00	0.00	0.02	0.01	0.01

architectures) and 16GB of RAM, under Ubuntu 16.04.1 LTS. The program used the LP solver from the CPLEX 12.6.3 Callable Library (with default settings). More specifically, we used primal simplex to solve the initial relaxation and dual simplex to re-optimize after adding cuts.

Finally, we ran the cutting-plane algorithm again for each instance, switching on and off various combinations of the cuts (35)–(37). The purpose of this was to enable us to identify the cuts that tend to be most useful in practice.

6.3 Results

We present results for six versions of the cutting-plane algorithm:

- “ \emptyset ”: Constraints (34) alone, without any Bi-P cuts.
- “T1”: Constraints (34) plus type-1 Bi-P cuts (35).
- “T2”: Type-2 Bi-P cuts (36) alone.
- “MC”: Constraints (34) plus MC cuts (37).
- “T1+T2”: Type-1 cuts (35) and type-2 cuts (36).
- “All”: Type-1 cuts (35), type-2 cuts (36) and MC cuts (37).

Tables 1 and 2 show, for each set of 10 instances and each combination of cutting planes, the average gap between the upper bound from the cutting-plane algorithm and the lower bound from the exact algorithm, expressed as a percentage of the lower bound.

From the tables, we see that the unstrengthened cuts (34) lead to extremely poor upper bounds in all cases. Moreover, the gaps for $|J| = 4$ are much worse than the gaps for $|J| = 6$. Type-1, type-2 and MC cuts all close the gap considerably, but type-1 and type-2 cuts appear to be more effective than MC cuts. Using type-1 and type-2 cuts in combination is particularly effective. In fact, the benefit gained by including MC cuts as well is rather small.

Table 2: Average percentage gaps when $|J| = 6$

Cuts	Demand Ratio				
	0.75	0.80	0.85	0.90	0.95
\emptyset	275	276	279	286	291
T1	0.08	0.08	0.14	0.15	0.05
T2	0.01	0.01	0.04	1.64	0.81
MC	2.30	2.41	1.63	0.33	0.15
T1, T2	0.01	0.01	0.04	0.15	0.03
All	0.01	0.01	0.04	0.14	0.03

7 Concluding Remarks

Perspective reformulations and cuts are an invaluable tool for both fractional programming and MINLP with indicator variables. We have shown that, when one is dealing with a mixed-integer fractional program with indicator variables, one needs to use “bi-perspective” reformulations and cuts in order to obtain bounds that are useful within an exact solution algorithm. We believe that extensions of perspective reformulations and cuts to other classes of problems would be a valuable topic for future research.

As for our specific application, to optimisation in OFDMA systems, we plan to look next at *stochastic dynamic* variants of the problem, in which users arrive and depart at random over time.

Acknowledgement: We are grateful to an anonymous referee whose careful reading and detailed suggestions enabled us to improve the paper substantially.

References

- [1] P. Bonami, A. Lodi, A. Tramontani & S. Wiese (2015) On mathematical programming with indicator constraints. *Math. Program.*, 115, 191–223.
- [2] S. Ceria & J. Soares (1999) Convex programming for disjunctive convex optimization. *Math. Program.*, 86, 595–614.
- [3] A. Charnes & W. Cooper (1962) Programming with linear fractional functionals. *Nav. Res. Log. Quart.*, 9, 181–186.
- [4] P.L. Combettes (2018) Perspective functions: properties, constructions, and examples. *Set-Valued Var. Anal.*, 26, 247–264.
- [5] K. Fazel & S. Kaiser (2008) *Multi-Carrier and Spread Spectrum Systems (2nd edn.)*. Chichester: Wiley.

- [6] A. Frangioni & C. Gentile (2006) Perspective cuts for a class of convex 0-1 mixed-integer programs. *Math. Program.*, 106, 225–236.
- [7] O. Günlük & J. Linderoth (2011) Perspective reformulation and applications. In: J. Lee & S. Leyffer (eds.) *Mixed Integer Nonlinear Programming*, pp. 61–89. Berlin: Springer.
- [8] S. Haykin (1994) *Communication Systems*. New York: Wiley.
- [9] J.-B. Hiriart-Urruty & C. Lémaréchal (1993) *Convex Analysis and Minimization Algorithms*. New York: Springer US.
- [10] J.E. Kelley, Jr. (1960) The cutting-plane method for solving convex programs. *SIAM J.*, 8, 703–712.
- [11] K. Kim, Y. Han & S.L. Kim (2005) Joint subcarrier and power allocation in uplink OFDMA systems. *IEEE Commun. Lett.*, 9, 526–528.
- [12] A.N. Letchford, Q. Ni & Z. Zhong (2017) An exact algorithm for a resource allocation problem in mobile wireless communications. *Comput. Optim. Appl.*, 68, 193–208.
- [13] A.N. Letchford, Q. Ni & Z. Zhong (2018) A heuristic for maximising energy efficiency in OFDMA systems. In J. Lee, G. Rinaldi & A.R. Mahjoub (eds.) *Combinatorial Optimization: 5th International Symposium*, pp. 303–312. Berlin: Springer.
- [14] Y.F. Liu & Y.H. Dai (2014) On the complexity of joint subcarrier and power allocation for multi-user OFDMA systems. *IEEE Trans. Signal Process.*, 62, 583–596.
- [15] Z.Q. Luo & S. Zhang (2008) Dynamic spectrum management: complexity and duality. *IEEE J. Sel. Topics Signal Process.*, 2, 57–73.
- [16] M. Resende & P. Pardalos (eds.) (2007) *Handbook of Optimization in Telecommunications*. New York: Springer US.
- [17] R.T. Rockafellar (1970) *Convex Analysis*. Princeton: Princeton University Press.
- [18] H.S. Ryoo & N.V. Sahinidis (1996) A branch-and-reduce approach to global optimization. *J. Glob. Optim.*, 8, 107–138.
- [19] S. Schaible (1983) Fractional programming. *Zeitschrift für Oper. Res.*, 27, 39–54.
- [20] K. Seong, M. Mohseni & J. Cioffi (2006) Optimal resource allocation for OFDMA downlink systems. In *Proc. 2006 IEEE Inter. Symp. Inform. Th.*, pp. 1394–1398.

- [21] C. Shannon (1949) Communication in the presence of noise. *Proc. IRE*, 1, 10–21.
- [22] T. Ting, S.F. Chien, X.S. Yang & S. Lee (2014) Analysis of quality-of-service aware orthogonal frequency division multiple access system considering energy efficiency. *IET Commun.*, 8, 1947–1954.
- [23] X. Xiao, X. Tao & J. Lu (2013) QoS-aware energy-efficient radio resource scheduling in multi-user OFDMA systems. *IEEE Commun. Lett.*, 17, 75–78.
- [24] C. Xiong, G. Li, S. Zhang, Y. Chen & S. Xu (2011) Energy-and spectral-efficiency tradeoff in downlink OFDMA networks. *IEEE Trans. Wirel. Commun.*, 10, 3874–3886.
- [25] W. Yu & R. Lui (2006) Dual methods for nonconvex spectrum optimization of multicarrier systems. *IEEE Trans. Commun.*, 54, 1310–1322.
- [26] C.C. Zarakovitis C & Q. Ni (2016) Maximising energy efficiency in multi-user multi-carrier broadband wireless systems: convex relaxation and global optimisation techniques. *IEEE Trans. Veh. Technol.*, 65, 5275–5286.