ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
"Antonio Ruberti"
CONSIGLIO NAZIONALE DELLE RICERCHE

A. Frangioni, C. Gentile

A COMPUTATIONAL COMPARISON OF REFORMULATIONS OF THE PERSPECTIVE RELAXATION: SOCP VS. CUTTING PLANES

R. 08-01, 2008

Antonio Frangioni – Dipartimento di Informatica, Polo Universitario della Spezia, Via dei Colli 90, 19121 La Spezia – Italy. Email: `frangio@di.unipi.it`.

Claudio Gentile – Istituto di Analisi dei Sistemi ed Informatica, CNR, Viale Manzoni 30 - 00185 Roma, Italy. Email: `gentile@iasi.cnr.it` Ph. +39 06 77161, Fax +39 06 7716461..

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", CNR

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: iasi@iasi.cnr.it
URL: http://www.iasi.cnr.it

Abstract

The *Perspective Reformulation* is a general approach for constructing tight approximations to MINLP problems with semicontinuous variables. Two different reformulations have been proposed for solving it, one resulting in a Second-Order Cone Program, the other based on representing the perspective function by (an infinite number of) cutting planes. We compare the two reformulations on two sets of MIQPs to determine which one is most effective in the context of exact or approximate Branch-and-Cut algorithms.

Key words: Mixed-Integer Non Linear Programs, Reformulations, Second-Order Cone Programs, Valid

Inequalities, Unit Commitment problem, Portfolio Optimization

1. Introduction

Semi-continuous variables are very often found in models of real-world problems such as production planning problems [17, 5, 7, 8], financial trading and planning problems [10, 6], and many others [4, 1, 9]. These are variables that are constrained to either assume the value 0, or to lie in some given *convex* compact set $\mathcal{P} \subseteq \mathbb{R}^m$; in our applications $\mathcal{P}$ will always be a polyhedron. Often $0 \notin \mathcal{P}$. For example, this the case when the variable represents the output of a production process that has a "nonzero minimum producible amount", but that can be switched off altogether. Alternatively, 0 may belong to $\mathcal{P}$, but one may incur in a fixed cost c to "activate" the process (produce a nonzero amount).

We will consider optimization with n semi-continuous variables $p_i \in \mathbb{R}^{m_i}$ for each $i \in N = \{1, \ldots, n\}$. Assuming that each $\mathcal{P}_i = \{p_i : A_i p_i \leq b_i\}$ has the property that $\{p_i : A_i p_i \leq 0\} = \{0\}$, each p_i can be modeled by using an associated binary variable u_i. We will consider Mixed-Integer NonLinear Programs (MINLP) of the form

$$\min g(z) + \textstyle\sum_{i \in N} f_i(p_i) + c_i u_i \tag{1}$$

$$A_i p_i \leq b_i u_i \qquad i \in N \tag{2}$$

$$(p, u, z) \in \mathcal{O} \tag{3}$$

$$u \in \{0,1\}^n \quad , \quad p \in \mathbb{R}^m \quad , \qquad z \in \mathbb{R}^q \tag{4}$$

where all f_i and g are closed convex functions, z is the vector of the non-semi-continuous variables, and $\mathcal{O}$ is any subset of $\mathbb{R}^{m+n+q}$ (with $m = \sum_{i \in N} m_i$), representing the other constraints of the problem.

It is known that the convex hull of the (disconnected) domain $\{0\} \cup \mathcal{P}_i$ of each p_i can be conveniently represented in a higher-dimensional space, which allows the derivation of *disjunctive cuts* for the problem [14]. This leads to defining the *Perspective Reformulation* (PRef) of (MINLP) [3, 5]

$$\min g(z) + \textstyle\sum_{i \in N} u_i f_i(p_i/u_i) + c_i u_i \tag{5}$$

$$(2) \quad , \quad (3) \quad , \quad (4) \quad .$$

While (5) is undefined when some $u_i = 0$, it can be extended by continuity to allow for null values. This results in (5) and (1) coincident for $u \in \{0,1\}^n$, hence (PRef) is a "good" reformulation of (MINLP) since its continuous relaxation, called the *Perspective Relaxation* (PRel), provides significantly stronger bounds than the continuous relaxation of (MINLP) [5, 6, 1, 8, 9]. We remark that $u_i f_i(p_i/u_i)$ for $u_i \geq 0$ is the *perspective function* of $f_i(p_i)$, a well-known tool in convex analysis, hence the name.

However, an issue with (PRel) is the high nonlinearity in the objective function due to the added fractional term. Two workable reformulations of (PRel) have been proposed: one as a Second-Order Cone Program (SOCP) [15, 1, 9], and the other as a Semi-Infinite Linear Program [5]. These are recalled in Section 2. In Section 3 we compare them, from a computational standpoint, in the context of exact or approximate Branch&Cut algorithms for two different Mixed-Integer Quadratic Programs (MIQP): the Mean-Variance problem (§ 3.1) and the Unit Commitment problem (§ 3.2), respectively.

2. The solution methods

2.1. SOCP reformulation

It is well-known that the epigraphs of many convex functions can be represented by means of *conic inequalities*; this is in particular true for the perspective function of any *SOCP-representable* convex function [2]. It is therefore not surprising that (PRel) can be written as a SOCP, as recently proposed in [1, 9] following suggestions dating back to [15], *provided that the same is possible for* (MINLP). The reformulation of (PRel) as a SOCP is actually quite simple in the quadratic case $f_i(p_i) = a_i p_i^2 + b_i p_i$, as when $u_i > 0$ a constraint $t_i \geq a_i p_i^2 / u_i$ can be algebraically transformed into the equivalent $(t_i +$

$u_i)^2/4 \geq a_i p_i^2 + (t_i - u_i)^2/4$, leading to the Mixed-Integer SOCP

$$\begin{array}{ll} \min & g(z) + \sum_{i \in N} t_i + b_i p_i + c_i u_i \\ & \sqrt{a_i p_i^2 + (t_i - u_i)^2/4} \leq (t_i + u_i)/2 \qquad i \in N \\ & (2) \ , \ (3) \ , \ (4) \ , \ t \in \mathbb{R}^n_+ \ , \end{array}$$

which can be approached with solvers such as `Cplex`. This can be more efficient than attacking (MINLP) directly [1, 9]. We call the above the Conic Program (CP) reformulation.

2.2. Perspective Cuts

An alternative formulation [5] is based on the fact that the epigraph of $uf(p/u) + cu$ on $conv(\ \{0\} \cup \mathcal{P}\ )$ can be represented by the following (infinite) family of linear inequalities, called *Perspective Cuts* (P/C),

$$v \geq sp + (c + f(\bar{p}) - s\bar{p})u \tag{6}$$

indexed over all the (uncountably many) $\bar{p} \in \mathcal{P}$ and $s \in \partial f(\bar{p})$, where $\partial f(\bar{p})$ denotes the subdifferential of f at $\bar{p}$. When f is quadratic, this leads to the following Semi-Infinite MINLP

$$\begin{array}{ll} \min & g(z) + \sum_{i \in N} v_i \\ & v_i \geq (2a_i\bar{p}_i + b_i)p_i + (c_i - a_i\bar{p}_i^2)u_i \qquad \bar{p}_i \in \mathcal{P}_i \ \ i \in N \\ & (2) \quad , \quad (3) \quad , \quad (4) \quad , \quad v \in \mathbb{R}^n \quad , \end{array}$$

which we call the P/C formulation of (PRef). While this problem cannot be solved directly, it lends itself nicely to iterative approximation techniques whereby a (small) finite subset of the P/C (6) are kept, the current solution (p^*, u^*, v^*) of the relaxation is produced, and all the violated P/C with $\bar{p}_i = p_i^*/u_i^*$ (assuming $0/0 = 0$) are added. This procedure can easily be implemented by using the standard tools made available by off-the-shelf solvers such as `Cplex`. Again, this is usually more efficient than approaching (MINLP) directly [5, 6, 8].

2.3. Features comparison

The two formulations have different potential strengths and weaknesses. CP is more appealing because it can be solved one-shot, instead of requiring a—theoretically, infinite—iterative process. However, it can only be used if the f_is are *SOCP-representable*, at least approximately [13]. Furthermore, SOCP-representing a function typically requires the introduction of auxiliary variables, whose number, roughly speaking, grows as the function becomes "more complex". Finally, conic programs require interior-point solution methods, which are less efficient than active-set ones in the context of enumerative approaches [16]. On the contrary, the P/C formulation can be used even if the f_is are *not SOCP-representable*, it always requires only *one* additional variable v_i for each $i \in N$, irrespective of the "complexity" of f_i, and (PRel) is a LP or QP if g and $\mathcal{O}$ are "simple enough", allowing to use more reoptimization-friendly active-set methods. Of the other hand, repeated solutions of the approximated versions of (PRel) are needed. Furthermore, if g and $\mathcal{O}$ are nonlinear then interior-point approaches may need to be used also for P/C, possibly negating it a potential advantage.

In the following, we will compare CP and P/C on the case where (MINLP) is a MIQP. This allows both approaches to be implemented within the same general-purpose solver, making the comparison between them as fair as possible. Besides, this is in some sense the "best case" for both approaches: in CP it only require *one* extra variable for each i, thus resulting in the smallest (all the rest being equal) formulation, and in P/C it allows the use of active-set solvers.

2.4. Implementation details

For our experiments we have used `Cplex 11`, which allows to directly input the CP formulation as a Mixed-Integer Quadratically Constrained Quadratic Program (QCQP). As for the P/C formulation, the dynamic generation of (6) can be easily implemented by means of the `cutcallback` procedure. Thus, apart from the basic formulation, the same sophisticated tools (valid inequalities, branching rules, ...)

are used for both. A few differences remain: for instance, the need for invoking the callback functions disables the—allegedly—more efficient *dynamic search* of `Cplex 11` for P/C, whereas it is used with CP. Apart from these, the very same machinery is used with both formulations, allowing a fair comparison.

The tests have been performed on an Opteron 246 (2 GHz) computer with 2 GigaBytes of RAM, running Linux Fedora Core 3. Unless otherwise stated, the default required gap for Mixed-Integer programs (0.01%) has been set; a maximum time limit of 24 hours (86400 seconds) of CPU time has been set.

3. Computational results

3.1. Markowitz Mean-Variance model

A set of n risky assets are available for purchase; for each asset i, the expected unit return μ_i for the considered time horizon is known, and *minimum and maximum buy-in thresholds* $0 < p_i^{min} < p_i^{max}$ are set on the purchasable quantity. The Mean-Variance (MV) model with minimum buy-in thresholds in portfolio optimization

$$\begin{array}{ll} \min & p^T Qp \\ & ep = 1 \quad , \quad \mu p \geq \rho \quad , \quad u \in \{0,1\}^n \\ & u_i p_i^{min} \leq p_i \leq u_i p_i^{max} \quad i \in N \quad , \end{array} \tag{7}$$

where $Q \succeq 0$ is the $n \times n$ variance-covariance matrix and e is the all-ones vector, requires the selection of a minimum-risk (as measured by variance) portfolio produceing a desired level of return ρ. This MIQP has a very "simple" structure, consisting almost only of the nonlinear semicontinuous variables; however, it does not directly qualify for (PRef), as the cost function is nonseparable. This can be dodged with a reformulation trick first proposed in [5], and somewhat reminiscent of the so-called *Larangian Decomposition*; compute a diagonal matrix $D \succeq 0$ such that $R = Q - D \succeq 0$, change the objective function to $p^T Dp + z^T Rz$, and add the additional constraint $z = p$. In this way, the perspective reformulation can be applied to the—now, separable—p variables, while all the "nonseparability" in the objective function is moved to the "other" variables z. An efficient and effective way for computing a "large" D is by solving a single SemiDefinite Program [6].

We have compared P/C and CP on 90 randomly generated MV instances, described in [6] and freely available at

`http://www.di.unipi.it/optimize/Data` .

The instances are characterized by the value of $n \in \{200, 300, 400\}$, and by the *dominance index* of Q, i.e., the average over all $i \in N$ of $1 - \sum_{j \neq i} |Q_{ij}|/Q_{ii}$, measuring how much the matrix is diagonally dominant; this turned out to have a significant impact on the effectiveness of the (PRef) [6]. The "+", "0" and "−" instances have, respectively, strongly, weakly, and strongly *not* diagonally dominant Q (the dominance index is ≈ 0.6, ≈ 0 and ≈ -0.5, respectively). For each combination, 10 instances are generated.

In Table 1 we report results of four different variants. For P/C, we have tested both with default `Cplex` settings, which lead to using the *quadratic simplex* for solving the relaxations during the B&C, as well as with forcing `Cplex` to use its IP algorithm throughout all the search. For CP, we have tested both with default `Cplex` settings and with `miqcpstrat = 2`, which implements a linearization-based method for the solution of QCQPs (new to `Cplex 11`) akin to [11, 12, 17]. In the Table, columns "nds" and "time" report the number of nodes in the B&C tree and the total running time (in seconds) required by each approach, while column "gap" reports, only for those cases where not all the instances could be solved to optimality within the allotted time limit, the attained gap (in percentage) at termination. The number in parenthesis next to the gap is the number of unsolved instances. Columns "LPs" (resp. "QPs", "CPs") and "t/LP" (resp. "t/QP", "t/CP") report respectively the total number of Linear (resp. Quadratic, Conic) Programs solved, and the average time required for solving one of them.

The results clearly favor P/C over CP. Using the default quadratic simplex allows extremely quick reoptimization, and therefore enumeration of enough B&C nodes to solve even the largest instances. Using the IP algorithm instead often has a significant positive effect on the number of explored nodes. The reason is not very clear; apparently, the "more interior" solutions it generates help the branching rules to perform better. However, since the cost per relaxation can be more than *two orders of magnitude*

Table 1: Results for MV

	P/C				P/C-IP					CP					CP - `miqcpstrat = 2`				
n	nds	QPs	time	t/QP	nds	QPs	time	t/QP	gap	nds	CPs	time	t/CP	gap	nds	LPs	time	t/LP	gap
200^+	1.9e4	1.9e4	194	0.0008	8.6e3	1.0e4	264	0.037		9.2e3	1.1e3	17961	1.578	0.15(1)	4.8e5	4.4e5	9264	0.027	
200^0	1.7e4	1.8e4	90	0.0007	1.1e5	1.3e5	348	0.030		2.7e4	3.2e4	30785	1.648	0.32(2)	9.8e5	1.0e6	70024	0.041	1.03(7)
200^-	1.2e5	1.3e5	835	0.0006	7.9e3	1.2e4	3815	0.031		1.6e4	1.9e5	55144	1.719	1.02(5)	2.0e6	1.9e6	78424	0.081	3.33(9)
300^+	3.4e4	3.5e4	433	0.0014	1.0e4	1.2e4	1946	0.163		1.1e4	1.4e4	72075	8.334	0.58(7)	7.8e5	7.0e5	31084	0.054	0.01(1)
300^0	3.1e5	3.3e4	378	0.0019	4.2e4	5.0e4	1635	0.083		1.0e4	1.3e4	59591	4.464	0.53(6)	7.2e5	7.2e5	69495	0.066	1.32(7)
300^-	5.5e5	5.8e4	654	0.0014	1.9e4	2.2e4	3955	0.076		1.1e4	1.3e4	66863	5.272	0.81(7)	1.1e6	1.1e6	65489	0.080	2.30(7)
400^+	7.9e4	8.2e4	2066	0.0032	2.1e4	2.5e4	39136	1.214	0.23(4)	4.7e3	5.9e3	61810	10.397	1.01(6)	7.6e5	7.7e5	54382	0.094	0.70(5)
400^0	2.3e5	2.4e5	3974	0.0020	1.9e5	2.2e5	22635	0.223	0.08(1)	6.1e3	7.6e3	83782	10.588	1.79(9)	6.4e5	6.5e5	86400	0.106	2.28(10)
400^-	3.3e5	3.4e5	8092	0.0026	8.8e4	1.0e5	44167	0.213	0.18(3)	6.3e3	7.9e3	80382	10.764	2.71(8)	8.2e5	8.4e5	86400	0.135	3.77(10)

higher, P/C-IP is never competitive with P/C. It is instead quite competitive with CP, which requires a comparable (often slightly smaller) number of nodes, but whose relaxation cost is even higher by *at least one order of magnitude*, often more. Using the linearization-based method provided by `Cplex` produces mixed results: the cost per relaxation does indeed decrease very significantly, although that of standard P/C is still considerably lower, but the number of B&C nodes, and especially the number of LPs, is significantly larger than in all other cases. The net result is that while the `miqcpstrat = 2` setting does improve on the results of standard CP for the "easy $^+$" instances, where the quality of the bound is better, it actually worsens them in all other cases. All in all, the P/C reformulation, especially when the quadratic simplex is used, is by far the more efficient one in this case.

3.2. The Unit Commitment problem

The Unit Commitment (UC) problem in electrical power production requires to optimally operate a set I of thermal generating units and a set H of hydro generating units to satisfy a given total power demand on each of a set T of discretized time instants, covering some time horizon (e.g., hours in a day or a week). Each thermal unit $i \in I$ is characterized by a minimum and maximum power output $0 < p_i^{min} < p_i^{max}$, when the unit is operational, and by a convex quadratic power (fuel) cost function $f_i(p) = a_i p^2 + b_i p + c_i$. Thus, UC is a MIQP with $n = |T| \cdot |I|$ semi-continuous variables. Besides these, the problem includes several other groups of variables and constraints. We will not describe the complete formulation here for space reasons; the interested reader is referred e.g. to [7, 8]. It is however worth mentioning that thermal units are subject to *minimum up- and down-time* and *ramp rate* constraints, hydro units are subject to *mass balance* and *reservoir volume* constraints, while the interconnecting electrical network adds *spinning reserve* and *capacity* constraints. What is relevant here is that UC problems have a "rich" structure, besides that of nonlinear-cost semicontinuous variables.

We have compared P/C and CP on a test bed of randomly generated realistic instances already employed in [5, 7, 8], and freely available at the previously mentioned web address. In the tables, "p" is number of thermal units (hence $n = 24p$, as the time discretization is hourly on daily instances) and "h" is the number of hydro units. The first half of the tables, with $h = 0$, is therefore composed by "pure thermal" instances, and each row reports averaged results of 5 instances of the same size.

Table 2: Results for UC with optimality tolerance 0.01%

				P/C					CP		
p	h	gap	nds	LPs	time	t/LP	gap	nds	CPs	time	t/CP
10	0		4.3e2	7.8e2	14	0.018		5.8e2	1.0e3	20	0.021
20	0		5.0e4	5.8e4	6805	0.094		6.6e4	7.5e4	13392	0.145
50	0	0.08	1.7e5	2.1e5	86400	0.421	0.08	9.1e4	1.1e5	86400	0.781
20	10		1.1e4	1.3e4	161	0.014		1.4e4	1.8e4	626	1.937
50	20		5.5e5	6.6e5	29874	0.037	0.00	5.0e5	6.1e5	86400	0.460
75	35	0.01	8.5e5	1.0e6	73076	0.073	0.01	1.8e5	2.2e5	86400	0.314

Table 2 reports the results with standard optimality tolerance 0.01%. These are limited to smaller-size instances, as *none* the approaches could solve *any* of the largest-size ones within the 24 hours time limit. The results confirm a distinct advantage of P/C over CP, but of a largely reduced magnitude. This is due to the fact that LPs are "only" up to two orders of magnitude faster to solve than CPs, most often less, as opposed to the 4+ orders of magnitude witnessed in the MV case. This is likely due to the fact that these instances have a much larger number of constraints and continuous variables, those devoted to modeling the hydro units, and IP approaches have a better asymptotic complexity than active-set ones which actually shows in practical performances. Indeed, relative performances of CP w.r.t. P/C seem to improve as the size of the instances grow. Also, P/C requires somewhat less nodes. Since the lower bound is the same, the difference is likely due to the fact that the corresponding formulation is a MILP, for which more algorithmic options (such as Gomory cuts) are available with respect to the MI-QCQP corresponding to CP. Specific tests, not reported here for space reasons, have excluded that the heuristics play a major role in this, as was the case in [7].

Table 3: Results for UC with optimality tolerance 0.5%

		P/C					CP				
p	h	gap	nds	LPs	time	t/LP	gap	nds	CPs	time	t/CP
10	0	0.09	0	30	0.67	0.023	0.06	0	70	1.91	0.028
20	0	0.06	0	34	2.81	0.085	0.09	0	65	6.78	0.106
50	0	0.18	0	39	15.45	0.411	0.19	0	91	37.91	0.421
75	0	0.22	0	30	23.28	0.785	0.23	0	71	63.27	0.933
100	0	0.15	0	29	34.16	1.182	0.19	0	64	100.28	1.578
150	0	0.10	0	75	90.13	1.410	0.11	0	106	233.46	2.256
200	0	0.09	0	57	126.28	2.313	0.11	0	104	386.36	3.860
20	10	0.11	0	83	2.77	0.034	0.24	20	194	10.45	2.372
50	20	0.04	0	79	6.53	0.102	0.35	1	115	20.55	0.575
75	35	0.09	0	61	10.60	0.182	0.08	15	202	64.50	0.319
100	50	0.04	0	81	20.17	0.267	0.08	10	193	97.03	0.421
150	75	0.06	100	417	247.73	0.596	0.04	15	331	368.92	0.778
200	100	0.04	30	222	247.22	1.111	0.03	5	165	385.03	1.563

In order to test the approaches on larger instances we also experimented with the much coarser optimality tolerance of 0.5%. This is the advised value for quickly obtaining approximated solutions when the operational environment requires fast response times [7, 8]. The corresponding results, reported in Table 3, confirm the previous analysis. All the pure thermal instances are solved at the root node by both reformulations. Despite the fact that P/C inherently requires repeated LP solutions due to the iterative nature of the approach, CP ends up actually requiring more relaxation solutions than P/C to construct a good feasible solution. LPs are still, on average, faster than CPs, although much less so than in the previous cases; this is due to the much smaller number of relaxations solved overall, which reduces the impact of reoptimization. The comparison between the two approaches is somewhat complicated by the fact that on hydro-thermal instances the two reformulations require a different amount of enumeration; however, overall P/C is about a factor of three faster than CP, and the quality of the obtained solution is most often slightly better. Interestingly, the `miqcpstrat = 2` setting was found in this case to be even less effective than for MV; the results are not reported here due to space concerns.

3.3. Conclusion

The Perspective Relaxation is a useful tool for obtaining tighter lower bounds on nonlinear programs with semicontinuous variables. Both the Conic Program and the Perspective Cut reformulation allow to exploit state-of-the-art, off-the-shelf solvers to compute them. Currently, the P/C reformulation seems to be favored, at least in the two applications that we tested. This is mostly due to the much more efficient reoptimization capabilities of active-set algorithms with respect to Interior Point ones. It should be remarked that P/C may be less competitive for "more nonlinear" problems than MIQPs, as discussed in §2.3, where the "other" structures of the problem (g, $\mathcal{O}$) are inherently conic. Also, our results suggest that the CP reformulation becomes more competitive as the size of the instances grows, and for instances with "rich" structure. However, even in that case the use of active-set LP technology should not be ruled out a priori. This has been recently shown in [16], where an efficient LP approximation of the Second-Order Cone (in a lifted space) is shown to outperform IP methods in the context of the solution of MI-SOCPs precisely because of the vastly superior reoptimization capabilities of the simplex method, despite the fact that IP methods are much better for the one-off solution of SOCPs. All in all, for the current state of solution technology, and on original formulations with linear constraints, the apparently more awkward P/C reformulation seems to have a computational edge over the more compact an elegant CP one.

References

[1] S. Aktürk, A. Atamtürk, and S. Gürel, "A strong conic quadratic reformulation for machine-job assignment with controllable processing times," *Operations Research Letters*, vol. to appear, 2009.

[2] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, Engineering Applications.* MPS-SIAM Series on Optimization, Philadelphia: SIAM, 2001.

[3] S. Ceria and J. Soares, "Convex programming for disjunctive convex optimization," *Mathematical Programming*, vol. 86, pp. 595–614, 1999.

[4] M. Duran and I. Grossmann, "An outer-approximation algorithm for a class of mixed-integer nonlinear programs," *Mathematical Programming*, vol. 36, pp. 307–339, 1986.

[5] A. Frangioni and C. Gentile, "Perspective Cuts for 0-1 Mixed Integer Programs," *Mathematical Programming*, vol. 106, no. 2, pp. 225–236, 2006.

[6] A. Frangioni and C. Gentile, "SDP Diagonalizations and Perspective Cuts for a Class of Nonseparable MIQP," *Operations Research Letters*, vol. 35, no. 2, pp. 181 – 185, 2007.

[7] A. Frangioni, C. Gentile, and F. Lacalandra, "Solving Unit Commitment Problems with General Ramp Contraints," *International Journal of Electrical Power and Energy Systems*, vol. 30, pp. 316 – 326, 2008.

[8] A. Frangioni, C. Gentile, and F. Lacalandra, "Tighter Appro-ximated MILP Formulations for Unit Commitment Problems," *IEEE Transactions on Power Systems*, vol. to appear, 2009.

[9] O. Günlük and J. Linderoth, "Perspective relaxation of MINLPs with indicator variables," in *Proceedings 13^{th} IPCO* (A. Lodi, A. Panconesi, and G. Rinaldi, eds.), vol. 5035 of *Lecture Notes in Computer Science*, pp. 1–16, 2008.

[10] N. Jobst, M. Horniman, C. Lucas, and G. Mitra, "Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints," in *Quantitative Finance*, vol. 1, pp. 1–13, Wiley, Chichester, 2001.

[11] A. Oĺafsson and S. Wright, "Efficient schemes for robust IMRT treatment planning," *Physics in Medicine and Biology*, vol. 51, pp. 5621–5642, 2006.

[12] I. Quesada and I. Grossmann, "An LP/NLP based branch-and-bound algorithm for convex MINLP optimization problems," *Computers and Chemical Engineering*, vol. 16, pp. 937–947, 1992.

[13] N. Sawaya and I. Grossmann, "Computational implementation of non-linear convex hull reformulation," *Computers & Chemical Engineering*, vol. 31, no. 7, pp. 856–866, 2007.

[14] R. Stubbs and S. Mehrotra, "A branch-and-cut method for 0-1 mixed convex programming," *Mathematical Programming*, vol. 86, pp. 515–532, 1999.

[15] M. Tawarmalani and N. Sahinidis, "Semidefinite relaxations of fractional programs via novel convexifications techniques," *Journal of Global Optimization*, vol. 20, pp. 137–158, 2001.

[16] J. Vielma, S. Ahmed, and G. Nemhauser, "A lifted linear programming branch-and-bound algorithm for mixed integer conic quadratic programs," *INFORMS Journal on Computing*, vol. 20, no. 3, pp. 438–450, 2008.

[17] J. Zamora and I. Grossmann, "A global MINLP optimization algorithm for the synthesis of heat exchanger networks with no stream splits," *Comput & Chem. Engin.*, vol. 22, pp. 367–384, 1998.