A counterexample to an exact extended formulation for the single-unit commitment problem☆

Tiziano Bacci[a,*], Antonio Frangioni[b], Claudio Gentile[c]

[a] *Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti", Consiglio Nazionale delle Ricerche, Italy*
[b] *Dipartimento di Informatica, Università di Pisa, Italy*
[c] *Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti", Consiglio Nazionale delle Ricerche, Italy*

Abstract

Recently, Guan, Pan, and Zohu presented a MIP model for the thermal single-unit commitment claiming that provides an integer feasible solution for any convex cost function. In this note we provide a counterexample to this statement and we produce evidence that the perspective function is needed for this aim.

Keywords: Unit Commitment, Mixed Integer Nonlinear Programming, Perspective Function, Exact formulation

1. Introduction

The Unit Commitment (UC) problem is a basic problem arising in power industries to coordinate and manage power generation units. Originated in now mostly bygone monopolistic regimes, research on UC have been ongoing for over 60 years and it still is very much an active area, with many hundredths of scientific articles (cf. e.g. [1]) and new ones appearing continuously.

Lately, a prolific trend has been the study of "tight" Mixed-Integer Programming formulations; in particular, the operating constraints of thermal units present an interesting combinatorial structure, and therefore provide a challenging research topic. Recently, three different groups have independently proposed "exact" MIP formulations for a quite general form of the *single-unit* (also known as *self-scheduling*) problem including minimum up- and down-time constraints, ramp constraints, start-up/shut-down limits, and history-dependent start-up costs. While single-unit problems are typically solved efficiently by

☆This paper has been supported by the Italian Government project PRIN2015B5F27W "Nonlinear and Combinatorial Aspects of Complex Networks" and by the European H2020 project MINOA GA764759 "Mixed Integer Nonlinear Optimization: Algorithms and Applications".

*Corresponding author

Email addresses: tiziano.bacci@iasi.cnr.it (Tiziano Bacci), frangio@di.unipi.it (Antonio Frangioni), gentile@iasi.cnr.it (Claudio Gentile)

Dynamic Programming (DP) algorithms even with convex nonlinear objective functions [2] (as required by many applications), exact self-scheduling models can clearly contribute to tighter formulations for the more challenging multi-unit versions. The first proposal has been [3], based on the exact DP algorithm of [2]; then, similar results with different proof techniques appeared in [4]. These results were for *linear* objective functions, and therefore resulted in Mixed-Integer Linear formulations. Recently, [5] presented a similar approach claiming that the proposed formulation is “exact” (its continuous relaxation always has integer optimal solutions) also for general convex costs. However, this claim is false. This note provides a counterexample for the case of simple convex quadratic costs, which are the most common costs used in power industry.

The structure of the note is the following: in Section 2 the formulation proposed in [5] is recalled. In Section 3 we show a counterexample to the statement that the model in [5] is exact for any convex cost function, and we clarify where the conceptual error is. Finally, in Section 4 we announce a companion paper partly solving the problem of finding an exact MINLP formulation for the case of convex cost functions, and in the Appendix we provide full LP-format files (read by most current general-purpose solvers) to replicate it.

2. The Guan-Pan-Zohu model

In [5] the following model is presented for the thermal single-unit commitement problem indicated there as (9a)-(9k):

$$\begin{aligned}
\min\ & \textstyle\sum_{t=1}^{T} SU(s_0+t-1)\alpha_t + \sum_{t=1}^{T}\sum_{k=t+L-1}^{T-1} SD(k-t+1)\beta_{tk} + \\
& \textstyle\sum_{t=L}^{T-\ell-1}\sum_{k=t+\ell+1}^{T} SU(k-t-1)\gamma_{tk} + \sum_{tk\in\mathcal{TK}}\sum_{s=t}^{k} w_{tk}^{s}(q_{tk}^{s},\beta_{tk}) && \text{(9a)}\\
\text{s.t.}\ & \textstyle\sum_{t=1}^{T}\alpha_t \le 1 && \text{(9b)}\\
& \textstyle -\alpha_t + \sum_{k=\min(t+L-1,T)}^{T}\beta_{tk} - \sum_{k=L}^{t-l-1}\gamma_{kt} = 0 \qquad t\in[1,T]_{\mathbb{Z}} && \text{(9c)}\\
& \textstyle -\sum_{k=1}^{t-L+1}\beta_{kt} - \sum_{k=\min(t+l+1,T)}^{T}\gamma_{tk} \le 0 \qquad t\in[L,T-l-1]_{\mathbb{Z}} && \text{(9d)}\\
& \textstyle \theta_t - \sum_{k=1}^{t-L+1}\beta_{kt} = 0 \qquad t\in[T-l,T]_{\mathbb{Z}} && \text{(9e)}\\
& \underline{C}\beta_{tk} \le q_{tk}^{s} \le \overline{C}\beta_{tk} \qquad s\in[t,k]_{\mathbb{Z}}\ ,\ \ tk\in\mathcal{TK} && \text{(9f)}\\
& q_{tk}^{t} \le \overline{V}\beta_{tk} \qquad s\in[t,k]_{\mathbb{Z}} && \text{(9g)}\\
& q_{tk}^{k} \le \overline{V}\beta_{tk} \qquad s\in[t,k]_{\mathbb{Z}}\ ,\ \ k\le T-1 && \text{(9h)}\\
& q_{tk}^{s-1} - q_{tk}^{s} \le V\beta_{tk} \qquad s\in[t+1,k]_{\mathbb{Z}}\ ,\ \ tk\in\mathcal{TK} && \text{(9i)}\\
& q_{tk}^{s} - q_{tk}^{s-1} \le V\beta_{tk} \qquad s\in[t+1,k]_{\mathbb{Z}}\ ,\ \ tk\in\mathcal{TK} && \text{(9j)}\\
& \alpha,\beta,\gamma \ge 0 && \text{(9k)}
\end{aligned}$$

The model uses the following parameters

T	length of the time horizon
L, ℓ	minimum up- and down-time
$\mathcal{TK}$	set of all possible pairs $t \in [1,T]_{\mathbb{Z}}$ and $k \in [min\{t+L-1,T\},T]_{\mathbb{Z}}$
SU, SD	start-up cost, shut-down cost
s_0	number of instants generator has been off before time 1 (assumed $s_0 \geq \ell$)
$w^s_{tk}(q^s_{tk}, \beta_{tk})$	production cost function
$\underline{C}$, $\overline{C}$	generation lower/upper bound
V	ramp-up/-down rate
$\overline{V}$	start-up/shut-down limits

and the following decision variables:

α_t	denotes that the first start-up occurs at time t;
β_{tk}	denotes that the unit is on from t to k;
γ_{tk}	denotes that the unit is off from $t+1$ to $k-1$;
θ_t	denotes that the unit is on at time t;
q^s_{tk}	denotes the power generated at time s if $\beta_{tk} = 1$.

Note that we have corrected constraints (9c) and (9d), corresponding to constraints (9c) and (9d) in [5], as the original form in [5] did not consider the right initial indices when the limits of the summations hit the time limit T. In [5], model (9a)–(9k) is determined as the dual of model (7a)–(7e), that is a Dynamic Programming model. Indeed, the model can be interpreted as minimum cost path problem on an acyclic graph with the following features:

- nodes u^+_k for each $k \in T$;
- nodes u^-_k for each $k \in T$;
- two additional nodes *source* and *sink*;
- arcs (u^-_t, u^+_k), associated with variables β_{tk}; when traversed by the flow they denote that the unit is on from t to k (extreme included);
- arcs (u^+_k, u^-_t), associated with variables γ_{kt}; when traversed by the flow they denote that the unit is off from k to t (extreme excluded);
- arcs $(source, u^-_t)$, associated with variables α_t, that denote that the unit will remain off until time $t-1$.

Indeed, (9c) are flow conservation constraints at nodes u^-_t while (9d) are flow conservation constraints at nodes u^+_t (except for a missing variable associated with the arc $(u^+_t, sink)$, that corresponds a slack). The other constraints can be interpreted very easily.

3. A counterexample

Here we describe a simple counterexample to Proposition 1 in [5]:

Proposition 1. *The optimal solution to the dual formulation (9)* (cf. here (9a)-(9k)) *is binary with respect decision variables α, β, γ, and θ.*

We point out that the counterexample is mainly based on the observation that the description of the convex hull of the feasible solutions is not sufficient to obtain the desired property when the objective function is nonlinear (even if convex, cf. e.g. [6]). Indeed, the counterexample considers only basic features of UC, that is, a convex quadratic objective function and minimum/maximum power limits, while relaxing all other features by setting high values for ramp rates and start-up/shut-down limits. In the example we set minimum up- and down-time equal to 3, but it is also possible to reduce this value to 1.

The instance.. We describe the simple instance that constitutes a counterexample:

- $T = 4$,
- $\underline{C} = 44$, $\overline{C} = 119$
- $s_0 = 5$
- $L = 3$, $\ell = 3$
- $SU(t) = 0$, $SD(t) = 0, t = 1, \ldots, T$,
- $V = \overline{C} - \underline{C} = 75$,
- $\overline{V} = \overline{C} = 119$,
- $w^s_{tk}(q^s_{tk}, \beta_{tk}) = a(q^s_{tk})^2 + b_s q^s_{tk} + c\beta_{tk}$,
  where $a = 0.009096$, $b_1 = -1.30768$, $b_2 = -1.42883$, $b_3 = -1.47917$, $b_4 = -1.54395$, $c = 120.599$.

The model.. We remark that start-up (9g) and shut-down (9h) limits are not necessary, as $\overline{V} = \overline{C}$. Similarly, the ramp constraints (9i) and (9j) are not necessary, as $V = \overline{C} - \underline{C}$. Therefore, the model reduces to:

$$\min \quad \sum_{t \in \mathcal{TK}} \sum_{s=t}^{k} \left(0.009096(q^s_{tk})^2 + b_s q^s_{tk} + 120.599\beta_{tk} \right) \tag{9a}$$

$$\text{s.t.} \quad \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 \leq 1 \tag{9b}$$

$$\beta_{13} + \beta_{14} - \alpha_1 = 0 \tag{9c[1]}$$

$$\beta_{24} - \alpha_2 = 0 \tag{9c[2]}$$

$$\beta_{34} - \alpha_3 = 0 \tag{9c[3]}$$

$$\beta_{44} - \alpha_4 = 0 \tag{9c[4]}$$

$$44\beta_{tk} \leq q^s_{tk} \leq 119 \qquad s \in [t,k]_{\mathbb{Z}} \ , \quad tk \in \mathcal{TK} \tag{9f}$$

$$\alpha, \beta, \gamma \geq 0 \tag{9k}$$

The optimal solution.. It is very simple to check with the help of a MIQP solver (such as CPLEX) that an optimal solution for the previous problem instance is:

$\alpha_1 = 0.360962$	$\beta_{13} = 0.171256$	$q^1_{13} = 20.379406$	$q^2_{24} = 24.708599$
$\alpha_2 = 0.207635$	$\beta_{14} = 0.189706$	$q^2_{13} = 20.379406$	$q^3_{24} = 24.708599$
$\alpha_3 = 0.215512$	$\beta_{24} = 0.207635$	$q^3_{13} = 20.379406$	$q^4_{24} = 24.708599$
$\alpha_4 = 0.215891$	$\beta_{34} = 0.215512$	$q^1_{14} = 22.575026$	$q^3_{34} = 25.645983$
	$\beta_{44} = 0.215891$	$q^2_{14} = 22.575026$	$q^4_{34} = 25.645983$
		$q^3_{14} = 22.575026$	$q^4_{44} = 25.690986$
		$q^4_{14} = 22.575026$	

with objective function value equal to -72.018180917. Note that the values of the variables α and β are not binary, and this already establishes the counterexample.

Additional remarks.. It may, however, be worth remarking that a fractional optimal solution occurs in the counterexample because the objective function is nonlinear. Indeed, if we modify the example simply by setting $a = 0$, any MILP solver would confirm that the optimal solution becomes

$$\alpha_1 = 1, \quad \beta_{14} = 1, \quad q^1_{14} = q^2_{14} = q^3_{14} = q^4_{14} = 119$$

with an optimal value of -202.99997. Hence, the solution is now integral, although not the optimal integral solution of the original problem. In fact, the optimal value clearly provides a lower bound on that of the UC, but this is worse than the bound provided by (9a)–(9k). We can confidently state that the solution is not optimal because we can construct a tighter model using the well-known *Perspective Reformulation* technique, first proposed in [6] (incidentally, motivated exactly by UC) for constructing "tight" formulations of Mixed-Integer NonLinear problems with semi-continuous variables with convex nonlinear costs. Applying the technique to the counterexample results in

$$\begin{array}{ll} \min & \sum_{t \in \mathcal{TK}} \sum_{s=t}^{k} \left(z^s_{tk} + b_s q^s_{tk} + 120.599 \beta_{tk} \right) \\ \text{s.t.} & (9b) \ , \ (9c[1]) \ , \ (9c[2]) \ , \ (9c[3]) \ , \ (9c[4]) \ , \ (9f) \ , \ (9k) \\ & 0.009096 (q^s_{tk})^2 \leq z^s_{tk} \beta_{tk} \qquad\qquad s \in [t,k]_{\mathbb{Z}} \ , \ \ tk \in \mathcal{TK} \end{array}$$

The last nonlinear constraints are rotated Second-Order Cone constraints (that current solvers are typically able to handle), and therefore describe a convex region. This represent the (epigraph of the) *perspective function*

$$p(\, q^s_{tk} \, , \, \beta_{tk} \,) = 0.009096 (q^s_{tk})^2 / \beta_{tk}$$

that can be proven to be the *convex envelope* (best possible convex approximation) of the nonconvex (nonlinear) function corresponding to fact that q^s_{tk} is a semi-continuous variable "governed" by the binary variable β_{tk}, i.e., $\beta_{tk} = 0 \implies q^s_{tk} = 0$, and $\beta_{tk} = 1 \implies q^s_{tk} \in [\, \underline{C} \, , \, \overline{C} \,]$. Indeed, with this change the formulation turns out to be “exact”: any appropriate solver (such as CPLEX) readily proves that its optimal solution is equal to zero for all variables, and therefore has optimal objective function equal to zero, too. The optimal solution being integral, is therefore optimal for the integer version of the problem as well (indeed, the obtained lower bound is much better than the previous two ones). This can be shown to be a general result.

4. Conclusions

In this note we have provided a counterexample for Proposition 1 in [5], stating that (9a)–(9k) is an exact formulation for the thermal single-unit commitment problem valid for any convex cost. We have also shown that the issue lies in the nonlinearity of the objective function, as the optimal solution indeed becomes integer if we make the objective linear. In order to get an integer solution in the nonlinear case, the *Perspective Reformulation* technique can be used for our example. In a companion paper [7], we will show that an “exact” formulation for the problem can always be obtained combining the two ingredients above, i.e., DP-based formulations like these of [3, 4, 5], and the Perspective Reformulation technique. The proof actually extends not only to Unit Commitment problems but to a large class of problems with analogous structure.

References

[1] W. van Ackooij, I. Danti Lopez, A. Frangioni, F. Lacalandra, M. Tahanan, Large-scale Unit Commitment under uncertainty: an updated literature survey, Annals of Operations Research 271 (1) (2018) 11–85.

[2] A. Frangioni, C. Gentile, Solving Nonlinear Single-Unit Commitment Problems with Ramping Constraints, Operations Research 54 (4) (2006) 767 – 775.

[3] A. Frangioni, C. Gentile, An extended MIP formulation for the single-unit commitment problem with ramping constraints, in: 17th British-French-German Conference on Optimization, London June 15-17, 2015.

[4] B. Knueven, J. Ostrowski, J. Wang, The ramping polytope and cut generation for the unit commitment problem, INFORMS Journal on Computing 30 (4) (2018) 739–749.

[5] Y. Guan, K. Pan, K. Zhou, Polynomial time algorithms and extended formulations for unit commitment problems, IISE Transactions 50 (8) (2018) 735–751.

[6] A. Frangioni, C. Gentile, Perspective cuts for a class of convex 0–1 mixed integer programs, Mathematical Programming 106 (2) (2006) 225–236.

[7] T. Bacci, A. Frangioni, C. Gentile, K. Tavlaridis-Gyparakis, New minlp formulations for the single-unit commitment problems with ramping constraints, Tech. Rep. 19-04, Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti", Consiglio Nazionale delle Ricerche (2019).

Appendix A. Complete LP file of the example

The following file in LP format, read by most general-purpose solvers, completely represents the counter-example.

```
Minimize
 obj: 361.797 B_1_3 - 1.30768 Q_1_3_1 - 1.42883 Q_1_3_2 - 1.47917 Q_1_3_3
      + 482.396 B_1_4 - 1.30768 Q_1_4_1 - 1.42883 Q_1_4_2 - 1.47917 Q_1_4_3
      - 1.54395 Q_1_4_4 + 361.797 B_2_4 - 1.42883 Q_2_4_2 - 1.47917 Q_2_4_3
      - 1.54395 Q_2_4_4 + 241.198 B_3_4 - 1.47917 Q_3_4_3 - 1.54395 Q_3_4_4
      + 120.599 B_4_4 - 1.54395 Q_4_4_4 + [ 0.018192 Q_1_3_1 ^2
      + 0.018192 Q_1_3_2 ^2 + 0.018192 Q_1_3_3 ^2 + 0.018192 Q_1_4_1 ^2
      + 0.018192 Q_1_4_2 ^2 + 0.018192 Q_1_4_3 ^2 + 0.018192 Q_1_4_4 ^2
      + 0.018192 Q_2_4_2 ^2 + 0.018192 Q_2_4_3 ^2 + 0.018192 Q_2_4_4 ^2
      + 0.018192 Q_3_4_3 ^2 + 0.018192 Q_3_4_4 ^2 + 0.018192 Q_4_4_4 ^2 ] / 2
Subject To
 _9b#0:         A_1 + A_2 + A_3 + A_4 <= 1
 _9c_1#1:       B_1_3 + B_1_4 - A_1  = 0
 _9c_2#2:       B_2_4 - A_2  = 0
 _9c_3#3:       B_3_4 - A_3  = 0
 _9c_4#4:       B_4_4 - A_4  = 0
 _9fa_1_3_1#5:  44 B_1_3 - Q_1_3_1 <= 0
 _9fa_1_3_2#6:  44 B_1_3 - Q_1_3_2 <= 0
 _9fa_1_3_3#7:  44 B_1_3 - Q_1_3_3 <= 0
 _9fa_1_4_1#8:  44 B_1_4 - Q_1_4_1 <= 0
 _9fa_1_4_2#9:  44 B_1_4 - Q_1_4_2 <= 0
 _9fa_1_4_3#10: 44 B_1_4 - Q_1_4_3 <= 0
 _9fa_1_4_4#11: 44 B_1_4 - Q_1_4_4 <= 0
 _9fa_2_4_2#12: 44 B_2_4 - Q_2_4_2 <= 0
 _9fa_2_4_3#13: 44 B_2_4 - Q_2_4_3 <= 0
 _9fa_2_4_4#14: 44 B_2_4 - Q_2_4_4 <= 0
 _9fa_3_4_3#15: 44 B_3_4 - Q_3_4_3 <= 0
 _9fa_3_4_4#16: 44 B_3_4 - Q_3_4_4 <= 0
 _9fa_4_4_4#17: 44 B_4_4 - Q_4_4_4 <= 0
 _9fb_1_3_1#18: - 119 B_1_3 + Q_1_3_1 <= 0
 _9fb_1_3_2#19: - 119 B_1_3 + Q_1_3_2 <= 0
 _9fb_1_3_3#20: - 119 B_1_3 + Q_1_3_3 <= 0
 _9fb_1_4_1#21: - 119 B_1_4 + Q_1_4_1 <= 0
 _9fb_1_4_2#22: - 119 B_1_4 + Q_1_4_2 <= 0
 _9fb_1_4_3#23: - 119 B_1_4 + Q_1_4_3 <= 0
 _9fb_1_4_4#24: - 119 B_1_4 + Q_1_4_4 <= 0
```

```
 _9fb_2_4_2#25: - 119 B_2_4 + Q_2_4_2 <= 0
 _9fb_2_4_3#26: - 119 B_2_4 + Q_2_4_3 <= 0
 _9fb_2_4_4#27: - 119 B_2_4 + Q_2_4_4 <= 0
 _9fb_3_4_3#28: - 119 B_3_4 + Q_3_4_3 <= 0
 _9fb_3_4_4#29: - 119 B_3_4 + Q_3_4_4 <= 0
 _9fb_4_4_4#30: - 119 B_4_4 + Q_4_4_4 <= 0
 _9g_1_3#31:    - 119 B_1_3 + Q_1_3_1 <= 0
 _9g_1_4#32:    - 119 B_1_4 + Q_1_4_1 <= 0
 _9g_2_4#33:    - 119 B_2_4 + Q_2_4_2 <= 0
 _9g_3_4#34:    - 119 B_3_4 + Q_3_4_3 <= 0
 _9g_4_4#35:    - 119 B_4_4 + Q_4_4_4 <= 0
 _9h_1_3#36:    - 119 B_1_3 + Q_1_3_3 <= 0
 _9i_1_3_2#37:  - 75 B_1_3 + Q_1_3_1 - Q_1_3_2 <= 0
 _9i_1_3_3#38:  - 75 B_1_3 + Q_1_3_2 - Q_1_3_3 <= 0
 _9i_1_4_2#39:  - 75 B_1_4 + Q_1_4_1 - Q_1_4_2 <= 0
 _9i_1_4_3#40:  - 75 B_1_4 + Q_1_4_2 - Q_1_4_3 <= 0
 _9i_1_4_4#41:  - 75 B_1_4 + Q_1_4_3 - Q_1_4_4 <= 0
 _9i_2_4_3#42:  - 75 B_2_4 + Q_2_4_2 - Q_2_4_3 <= 0
 _9i_2_4_4#43:  - 75 B_2_4 + Q_2_4_3 - Q_2_4_4 <= 0
 _9i_3_4_4#44:  - 75 B_3_4 + Q_3_4_3 - Q_3_4_4 <= 0
 _9j_1_3_2#45:  - 75 B_1_3 - Q_1_3_1 + Q_1_3_2 <= 0
 _9j_1_3_3#46:  - 75 B_1_3 - Q_1_3_2 + Q_1_3_3 <= 0
 _9j_1_4_2#47:  - 75 B_1_4 - Q_1_4_1 + Q_1_4_2 <= 0
 _9j_1_4_3#48:  - 75 B_1_4 - Q_1_4_2 + Q_1_4_3 <= 0
 _9j_1_4_4#49:  - 75 B_1_4 - Q_1_4_3 + Q_1_4_4 <= 0
 _9j_2_4_3#50:  - 75 B_2_4 - Q_2_4_2 + Q_2_4_3 <= 0
 _9j_2_4_4#51:  - 75 B_2_4 - Q_2_4_3 + Q_2_4_4 <= 0
 _9j_3_4_4#52:  - 75 B_3_4 - Q_3_4_3 + Q_3_4_4 <= 0
 _9k0_0#53:     A_1 >= 0
 _9k0_1#54:     A_2 >= 0
 _9k0_2#55:     A_3 >= 0
 _9k0_3#56:     A_4 >= 0
 _9k1_1_3#57:   B_1_3 >= 0
 _9k1_1_4#58:   B_1_4 >= 0
 _9k1_2_4#59:   B_2_4 >= 0
 _9k1_3_4#60:   B_3_4 >= 0
 _9k1_4_4#61:   B_4_4 >= 0
 _9k2_1_3#62:   G_1_3 >= 0
 _9k2_1_4#63:   G_1_4 >= 0
 _9k2_2_4#64:   G_2_4 >= 0
 _9k2_3_4#65:   G_3_4 >= 0
 _9k2_4_4#66:   G_4_4 >= 0
End
```

The results relative to the linear objective function can be obtained by replacing the objective function section in the file with

```
obj: 361.797 B_1_3 - 1.30768 Q_1_3_1 - 1.42883 Q_1_3_2 - 1.47917 Q_1_3_3
     + 482.396 B_1_4 - 1.30768 Q_1_4_1 - 1.42883 Q_1_4_2 - 1.47917 Q_1_4_3
     - 1.54395 Q_1_4_4 + 361.797 B_2_4 - 1.42883 Q_2_4_2 - 1.47917 Q_2_4_3
     - 1.54395 Q_2_4_4 + 241.198 B_3_4 - 1.47917 Q_3_4_3 - 1.54395 Q_3_4_4
     + 120.599 B_4_4 - 1.54395 Q_4_4_4
```

while those relative to the "perspectivized" exact formulation can be obtained by replacing the objective function with

```
obj: 361.797 B_1_3 - 1.30768 Q_1_3_1 - 1.42883 Q_1_3_2 - 1.47917 Q_1_3_3
     + 482.396 B_1_4 - 1.30768 Q_1_4_1 - 1.42883 Q_1_4_2 - 1.47917 Q_1_4_3
     - 1.54395 Q_1_4_4 + 361.797 B_2_4 - 1.42883 Q_2_4_2 - 1.47917 Q_2_4_3
     - 1.54395 Q_2_4_4 + 241.198 B_3_4 - 1.47917 Q_3_4_3 - 1.54395 Q_3_4_4
     + 120.599 B_4_4 - 1.54395 Q_4_4_4 + z_1_3_1  + z_1_3_2 + z_1_3_3
     + z_1_4_1 + z_1_4_2 + z_1_4_3 + z_1_4_4 + z_2_4_2 + z_2_4_3 + z_2_4_4
     + z_3_4_3 + z_3_4_4 + z_4_4_4
```

and adding the (rotated Second-Order Cone) constraints

```
[0.009096 Q_1_3_1 ^2 - B_1_3 * z_1_3_1 ] <= 0
[0.009096 Q_1_3_2 ^2 - B_1_3 * z_1_3_2 ] <= 0
[0.009096 Q_1_3_3 ^2 - B_1_3 * z_1_3_3 ] <= 0
[0.009096 Q_1_4_1 ^2 - B_1_4 * z_1_4_1 ] <= 0
[0.009096 Q_1_4_2 ^2 - B_1_4 * z_1_4_2 ] <= 0
[0.009096 Q_1_4_3 ^2 - B_1_4 * z_1_4_3 ] <= 0
[0.009096 Q_1_4_4 ^2 - B_1_4 * z_1_4_4 ] <= 0
[0.009096 Q_2_4_2 ^2 - B_2_4 * z_2_4_2 ] <= 0
[0.009096 Q_2_4_3 ^2 - B_2_4 * z_2_4_3 ] <= 0
[0.009096 Q_2_4_4 ^2 - B_2_4 * z_2_4_4 ] <= 0
[0.009096 Q_3_4_3 ^2 - B_3_4 * z_3_4_3 ] <= 0
[0.009096 Q_3_4_4 ^2 - B_3_4 * z_3_4_4 ] <= 0
[0.009096 Q_4_4_4 ^2 - B_4_4 * z_4_4_4 ] <= 0
```