Solving a general mixed-integer quadratic problem through convex reformulation : a computational study

Alain Billionnet, Sourour Elloumi, Amélie Lambert

Solving a general mixed-integer quadratic problem through convex reformulation : a computational study

Billionnet Alain, Elloumi Sourour, Lambert Amélie

Abstract. Let (QP) be a mixed integer quadratic program that consists of minimizing a quadratic function subject to linear constraints. In this paper, we present a convex reformulation of (QP), i.e. we reformulate (QP) into an equivalent program, with a convex objective function. Such a reformulation can be solved by a standard solver that uses a branch and bound algorithm. This reformulation, that we call `MIQCR (Mixed Integer Quadratic Convex Reformulation)`, is the best one within a convex reformulation scheme, from the continuous relaxation point of view. It is based on the solution of an SDP relaxation of (QP). Computational experiences were carried out with instances of (QP) with one equality constraint. The results show that most of the considered instances, with up to 60 variables, can be solved within 1 hour of CPU time by a standard solver.

1. Introduction

Consider the following linearly-constrained mixed-integer quadratic program:

$$(QP)\left\{\begin{array}{llll} Min & f(x) & & \\ s.t. & \displaystyle\sum_{i=1}^{N} a_{ri}x_i = b_r & r = \{1,\ldots,m\} & (1.1) \\ & \displaystyle\sum_{i=1}^{N} d_{si}x_i \leq e_s & s = \{1,\ldots,p\} & (1.2) \\ & 0 \leq x_i \leq u_i & i \in I & (1.3) \\ & 0 \leq x_i \leq u_i & i \in J & (1.4) \\ & x_i \in \mathbb{N} & i \in I & (1.5) \\ & x_i \in \mathbb{R} & i \in J & (1.6) \end{array}\right.$$

where $I = \{1,\ldots,n\}$ is the sub-set of integer variable indices, $J = \{n+1,\ldots,N\}$ is the sub-set of real variable indices,

$$f(x) = x^TQx + c^Tx = \sum_{(i,j)\in I^2} q_{ij}x_ix_j + \sum_{(i,j)\in I\times J} 2q_{ij}x_ix_j + \sum_{(i,j)\in J^2} q_{ij}x_ix_j + \sum_{i\in I\cup J} c_ix_i$$

and $Q \in \mathbf{S}_N$ (space of symmetric matrices of order N), $c \in \mathbb{R}^N$, $A \in \mathbf{M}_{m,N}$ (space of $m \times N$ matrices), $b \in \mathbb{R}^m$, $D \in \mathbf{M}_{p,N}$, $e \in \mathbb{R}^p$, $u \in \mathbb{N}^N$.

We suppose that the sub-function of the products of real variables of $h(x)$: $\sum_{(i,j)\in J^2} q_{ij}x_ix_j$ is convex.

(QP) belongs to the class of $\mathcal{NP}$-hard problems [3]. Standard solvers [6, 2] can efficiently solve Mixed Integer Quadratic Programs (MIQP), but only in the specific case where $f(x)$ is convex. Thus, to solve (QP) by use of a standard solver, we choose to reformulate it into another program with a convex objective function. By convex reformulation, we mean to design a program, that is equivalent to (QP), and that has a convex objective function. In concrete terms, that will consist of perturbing the Q matrix of $f(x)$ in order to obtain a positive semidefinite matrix.

In this work, we first define a convex reformulation scheme, and then we compute, within this scheme, the optimal convex reformulation in terms of continuous relaxation bound. To do it, we introduce new variables y_{ij}, and new linear constraints to enforce the equality $y_{ij} = x_ix_j$. These new variables will allow the perturbation of each term of matrix Q. In a sense, our approach mixes ideas of linearization and convexification.

In the rest of the paper, we present our approach, that we denote by `MIQCR (Mixed Integer Quadratic Convex Reformulation)`. In Section 2, we propose a reformulation scheme of (QP)

Keywords: Mixed integer quadratic programming, Convex reformulation, Semidefinite programming, Experiments.

into an equivalent mixed-integer quadratic program $(QP_{\alpha,\beta})$ depending on a scalar parameter α, and on a matrix parameter β. In Section 3, we show how to compute α^* and β^*, the values of α and β that maximize the optimal value of the continuous relaxation of $(QP_{\alpha,\beta})$. We show that α^* and β^* can be deduced from the solution of a semidefinite relaxation of (QP). Finally, in Section 4, we evaluate `MIQCR` from the computational point of view. Our experiments are carried out on instances of (QP) with one equality constraint. Section 5 is a conclusion.

2. A convex reformulation scheme for mixed-integer quadratic programs

In this section, we rewrite (QP) into an equivalent mixed-integer quadratic program $(QP_{\alpha,\beta})$ with a convex objective function. The idea is to add to the initial objective function $f(x)$ the following functions that vanish on the feasible domain of (QP) under the assumption that $y_{ij} = x_i x_j$, $\forall (i,j) \in P$, where $P = \{(i,j) \in (I \times I) \cup (I \times J) \cup (J \times I)\}$

- $\alpha \sum_{r=1}^{m} (\sum_{i \in I \cup J} a_{ri} x_i - b_r)^2$ where $\alpha \in \mathbb{R}$.
- $\sum_{(i,j) \in P} \beta_{ij}(x_i x_j - y_{ij})$, where $\beta_{ij} \in \mathbb{R}$ and $\beta_{ij} = \beta_{ji}$ $\forall (i,j) \in P$, or equivalently, we consider $\beta \in \mathbf{S}_N$ with $\beta_{ij} = 0$ $\forall (i,j) \in J^2$.

We obtain the following program $(QP_{\alpha,\beta})$:

$$(QP_{\alpha,\beta}) \begin{cases} Min & f_{\alpha,\beta}(x,y) \\ s.t. & (1.1)(1.2) \\ & x,y,z,t \in P_{xyzt} \end{cases}$$

where

$$f_{\alpha,\beta}(x,y) = f(x) + \sum_{(i,j) \in P} \beta_{ij}(x_i x_j - y_{ij}) + \alpha \sum_{r=1}^{m} (\sum_{i \in I \cup J} a_{ri} x_i - b_r)^2$$

and P_{xyzt} is the following set:

$$P_{xyzt} \left\{ x,y,z,t : \begin{array}{lll} (1.3)(1.4) & & \\ x_i = \displaystyle\sum_{k=0}^{\lfloor log(u_i) \rfloor} 2^k t_{ik} & i \in I & (2.1) \\ z_{ijk} \le u_j t_{ik} & (i,k) \in E,\ j \in I \cup J & (2.2) \\ z_{ijk} \le x_j & (i,k) \in E,\ j \in I \cup J & (2.3) \\ z_{ijk} \ge x_j - u_j(1 - t_{ik}) & (i,k) \in E,\ j \in I \cup J & (2.4) \\ z_{ijk} \ge 0 & (i,k) \in E,\ j \in I \cup J & (2.5) \\ y_{ij} = \displaystyle\sum_{k=0}^{\lfloor log(u_i) \rfloor} 2^k z_{ijk} & (i,j) \in I \times I \cup J & (2.6) \\ t_{ik} \in \{0,1\} & (i,k) \in E & (2.7) \\ y_{ij} = y_{ji} & (i,j) \in P & (2.8) \\ y_{ij} \ge x_i u_j + x_j u_i - u_i u_j & (i,j) \in P & (2.9) \\ y_{ii} \ge x_i & i \in I & (2.10) \\ y_{ij} \le u_i x_j & (i,j) \in I \times J & (2.11) \end{array} \right.$$

with $E = \{(i,k) : i = 1, \ldots, n,\ k = 0, \ldots \lfloor log(u_i) \rfloor\}$.

It is proven in [4] that $(QP_{\alpha,\beta})$ is equivalent to (QP).

3. Computing the best convex reformulation : the `MIQCR` method

In this section, we show how to compute, by semidefinite programming, values of α^* and β^* that make $f_{\alpha^*,\beta^*}(x,y)$ convex, and that maximize the continuous relaxation value of (QP_{α^*,β^*}), that is to say we have to solve the following problem (CP):

$$(CP) : \max_{\substack{\alpha \in \mathbb{R}, \beta \in \mathbf{S}_N \\ \beta_{ij}=0,\, (i,j) \in J^2 \\ Q_{\alpha,\beta} \succeq 0}} \{v(\overline{QP}_{\alpha,\beta})\}$$

where $(\overline{QP}_{\alpha,\beta})$ is the continuous relaxation of $(QP_{\alpha,\beta})$, $v(\overline{QP}_{\alpha,\beta})$ is the optimal solution value of $(\overline{QP}_{\alpha,\beta})$ and $Q_{\alpha,\beta} = Q + \alpha A^T A + \beta$. Recall that $\beta_{ij} = 0$, $\forall (i,j) \in J^2$.

Theorem 3.1. [4] *Let (SDP) be the following semidefinite program:*

$$
(SDP)\left\{
\begin{array}{lll}
Min & f(X,x) = \sum_{i=1}^{N}\sum_{j=1}^{N} q_{ij}X_{ij} + \sum_{i=1}^{N} c_i x_i & \\
s.t. & (1.1)(1.2)(1.4) & \\
 & \sum_{r=1}^{m}(\sum_{i=1}^{N}(\sum_{j=1}^{N} a_{ri}a_{rj}X_{ij} - 2a_{ri}b_r x_i)) = -\sum_{r=1}^{m} b_r^2 & (3.1)\\
 & X_{ij} \le u_j x_i \quad (i,j) \in P & (3.2)\\
 & X_{ij} \le u_i x_j \quad (i,j) \in P & (3.3)\\
 & -X_{ij} \le -u_j x_i - u_i x_j + u_i u_j \quad (i,j) \in P & (3.4)\\
 & -X_{ij} \le 0 \quad (i,j) \in P & (3.5)\\
 & -X_{ii} \le x_i \quad i \in I & (3.6)\\
 & \begin{pmatrix} 1 & x \\ x^T & X \end{pmatrix} \succeq 0 & (3.7)\\
 & x \in \mathbb{R}^N \quad X \in \mathbf{S}^N & (3.8)
\end{array}
\right.
$$

An optimal solution (α^*, β^*) of (CP) can be deduced from the optimal values of the dual variables of (SDP). The optimal coefficient α^* is the optimal value of the dual variable associated with constraint (3.1). The optimal coefficients β^*_{ij} are computed as $\beta^*_{ij} = \beta^{1*}_{ij} + \beta^{2*}_{ij} - \beta^{3*}_{ij} - \beta^{4*}_{ij}$, for $(i,j) \in P$, $i \neq j$, and $\beta^*_{ii} = \beta^{1*}_{ii} + \beta^{2*}_{ii} - \beta^{3*}_{ii} - \beta^{4*}_{ii} - \beta^{5*}_{ii}$, $i \in I$ where β^{1*}_{ij}, β^{2*}_{ij}, β^{3*}_{ij}, β^{4*}_{ij}, and β^{5*}_{ii}, are the optimal values of the dual variables associated with constraints (3.2), (3.3), (3.4), (3.5), and (3.6), respectively.

From Theorem 3.1, we design an exact solution algorithm for non-convex mixed-integer quadratic programs (QP) based on the `MIQCR` approach:

	Solution algorithm to (QP) based on `MIQCR`
1	Solve the semidefinite program (SDP)
2	Deduce α^* and β^* as in Theorem 3.1.
3	Solve the program (QP_{α^*,β^*}), by a MIQP solver. *(Its continuous relaxation $(\overline{QP}_{\alpha^*,\beta^*})$ is a convex program with an optimal value equal to the optimal value of (SDP))*

To illustrate our approach, we consider the following example.

Example 3.1. *Let (QP_e) be an instance of (QP) with 2 integer and 2 continuous variables:*

$$
(QP_e)\left\{
\begin{array}{lll}
Min & f(x) = x^T \left(\begin{array}{cc|cc} -7 & 3 & -15 & -4 \\ 3 & -14 & -7 & -13 \\ \hline -15 & -7 & 8 & 7 \\ -4 & -13 & 7 & 12 \end{array}\right) x + \begin{pmatrix} 15 \\ 10 \\ -7 \\ -4 \end{pmatrix}^T x & \\
s.t & 5x_1 + x_2 + 8x_3 + 4x_4 = 95 & \\
 & 0 \le x_i \le 10 & i \in \{1,\dots,4\} \\
 & x_1, x_2 \in \mathbb{N} & \\
 & x_3, x_4 \in \mathbb{R} &
\end{array}
\right.
$$

Observe that the sub-matrix $\begin{pmatrix} 8 & 7 \\ 7 & 12 \end{pmatrix}$ *is positive semidefinite.*
The optimal solution of (QP_e) is $x = (8, 10, 2.03, 7.19)$ and its value is -3434.27.
We perturb the Q matrix as follows:

$$
\begin{pmatrix}
-7+\beta_{11}+25\alpha & 3+\beta_{12}+5\alpha & -15+\beta_{13}+40\alpha & -4+\beta_{14}+20\alpha \\
3+\beta_{12}+5\alpha & -14+\beta_{22}+\alpha & -7+\beta_{23}+8\alpha & -13+\beta_{24}+4\alpha \\
-15+\beta_{13}+40\alpha & -7+\beta_{23}+8\alpha & 8+64\alpha & 7+32\alpha \\
-4+\beta_{14}+20\alpha & -13+\beta_{24}+4\alpha & 7+32\alpha & 12+16\alpha
\end{pmatrix}
$$

where the optimal value of the α parameter is 291.49, and the optimal values of the β parameter are: $\beta_{11} = \beta_{13} = \beta_{14} = 0$, $\beta_{12} = -8.73$, $\beta_{22} = 18.22$, $\beta_{23} = -0.005$, and $\beta_{24} = 4.16$.

For the reformulated problem, the optimal value of the continuous relaxation equals -3434.45*. The integrality gap is hence of* 0.005%*.*

4. Computational results

Our experiments concern instances of (QP) that consists of minimizing a quadratic function subject to a linear equality constraint.

$$(MQP)\begin{cases} Min & x^TQx + c^Tx \\ s.t. & \displaystyle\sum_{i=1}^{N} a_i x_i = b \\ & 0 \le x_i \le u_i \qquad i \in I \cup J \\ & x_i \in \mathbb{N} \qquad\qquad i \in I \\ & x_i \in \mathbb{R} \qquad\qquad i \in J \end{cases}$$

For this problem, we generate two classes of problems (MQP_1) and (MQP_2). For these two classes we randomly generate the coefficient u, Q, c, a and b in the same way, and we vary the number of integer variables. More precisely the coefficients of Q $\forall (i,j) \in P$ are integers uniformly distributed in the interval $[-100, 100]$ (for any $i < j$, a number ν is generated in $[-100, 100]$, and then $q_{ij} = q_{ji} = \nu$). To generate the coefficients of Q $\forall (i,j) \in J^2$, we generate a matrix $M \in \mathbf{M}_{N-n}$ of integers uniformly distributed in the interval $[-10, 10]$, and we compute $M' = M^T M$ then $q_{ij} = m'_{ij}$, $\forall (i,j) \in J^2$. The c coefficients are integers uniformly distributed in the interval $[-100, 100]$. The a_i coefficients are integers uniformly distributed in the interval $[1, 50]$, $b = 20 * \sum_{i=1}^{n} a_i$ and $u_i = 50$, $i \in I$. Note that in these instances the solution $x_i = 20$, for all i is feasible.

For the class (MQP_1), we take $1/3$ of integer variables, and $2/3$ of continuous ones. For the class (MQP_2), we take $2/3$ of integer variables, and $1/3$ of continuous ones.

For each problem and for each $N = 40$, 50, or 60, we generate 5 instances obtaining a total of 30 instances.

Our experiments are carried out on a PC with an Intel core 2 duo processor 2.8 GHz and 2048 MB of RAM using a Linux operating system. We use the modeling language ampl and the solver Cplex version 11 [2] for solving mixed integer quadratic convex programs, and the solver CSDP [1] for solving semidefinite programs.

Legends of the tables:

- *Name*: `Problem_i_r_nb`, where `i` is the number of integer variables in (QP), `r` is the number of real variables in (QP) and `nb` is the instance number.
- *Optimum*: best solution found within 1 hour of CPU time.
- *Initial gap*: $\left|\frac{opt - l}{opt}\right| * 100$ where l is the optimal value of the continuous relaxation at the root node.
- *CSDP time*: CPU time (in seconds) required by the SDP solver to find the optimal solution of the semidefinite relaxation of the initial problem.
- *Cplex time*: CPU time (in seconds) required by the branch-and-bound algorithm to solve the reformulated program. The time limit is fixed to 1 hour.
- *nodes*: number of nodes visited by the branch-and-bound algorithm.

Tables 1 and 2 present the results for the classes (MQP_1) and (MQP_2), respectively.

For the class (MQP_1), that has less integer variables than real ones, every instance of size 40, 50 or 60 can be solved by the MIQP solver in less than 142 seconds. For these instances the average initial gap over the 15 instances is of 0.13% and hence the number of nodes visited during the

		MIQCR			
Name	Optimum	initial gap	CSDP time(s)	Cplex time (s)	nodes
MQP_1_13_27_1	-1441722.34	0	1240	5	0
MQP_1_13_27_2	-4001173.84	0	1252	3	0
MQP_1_13_27_3	-4072930.12	0	1587	5	0
MQP_1_13_27_4	-4303086.42	0	1294	4	0
MQP_1_13_27_5	-5792796.84	0	1246	4	6
average		**0**	**1323.8**	**4.2**	**1.2**
MQP_1_16_34_1	-6770822.01	0	6088	10	0
MQP_1_16_34_2	-4281691.17	1.07	4746	57	58
MQP_1_16_34_3	-12003888.62	0	4747	17	7
MQP_1_16_34_4	-14917101.36	0	4726	11	0
MQP_1_16_34_5	-8450308.13	0.15	4744	42	51
average		**0.24**	**5010.2**	**27.4**	**23.2**
MQP_1_20_40_1	-11331739.74	0	14756	55	9
MQP_1_20_40_2	-9541493.46	0.20	17959	142	52
MQP_1_20_40_3	-11243727.11	0.37	14530	46	14
MQP_1_20_40_4	-17871860.96	0	14682	38	0
MQP_1_20_40_5	-11194683.38	0.15	15835	61	16
average		**0.14**	**15552.4**	**68.4**	**16.2**

Table 1. Solution of MQP_1

		MIQCR			
Name	Optimum	Initial gap	CSDP time(s)	Cplex time (s)	nodes
MQP_2_27_13_1	-12764814.30	1.53	4698	331	632
MQP_2_27_13_2	-13063090.09	0.73	4703	91	99
MQP_2_27_13_3	-12210409.85	2.72	4720	234	390
MQP_2_27_13_4	-15060832.30	0.80	4691	131	268
MQP_2_27_13_5	-11550064.15	0.83	4714	91	91
average		**1.32**	**4705.2**	**175.6**	**296**
MQP_2_34_16_1	-15746064.71	0.80	19164	1626	3088
MQP_2_34_16_2	-21504640.72	0.83	19100	104	0
MQP_2_34_16_3	-16337803.32	2.36	19245	1541	1349
MQP_2_34_16_4	-17942418.38	2.05	19174	1637	3973
MQP_2_34_16_5	-21394688.60	0.49	20327	646	210
average		**1.31**	**19402**	**1180.8**	**1724**
MQP_2_40_20_1	-34437235.29	0.08	54539	838	38
MQP_2_40_20_2	-26342341.28	2.29	54701	-	829
MQP_2_40_20_3	-25124557.73	6.61	54656	-	635
MQP_2_40_20_4	-27395752.85	11.02	54826	-	873
MQP_2_40_20_5	-22573097.16	8.69	54817	-	1038
average		**5.73**	**54707.8**	**838 (1)**	**682.6**

Table 2. Solution of MQP_2

Branch & Bound algorithm after reformulation is rather small, with an average of 13.5 nodes. Experiments on this class of problems give good first results.

To experiment the impact of the ratio of integer variables on the MIQCR approach, we generate a second class of instances, (MQP_2), in the same way that for (MQP_1), but we invert the ratio of integer variables versus the real one. The results reveal a similar trend. However, for this class of problems, MIQCR leads to a reformulated problem with bounds obtained by continuous relaxation a bit worst in comparison to the class (MQP_1). Indeed, the average initial gap increases from 0.13%

for (MQP_1) to 2.79% for (MQP_2). Consequently, the number of nodes visited in the Branch & Bound algorithm is then increased, with an average of 900.8 nodes.

Finally, let us mention the preprocessing time associated with the optimal solution of the semidefinite programs. For the class (MQP_1) it takes 1323.8 seconds, 5010.2 seconds and 15552.4 seconds on average, for instances of size 40, 50, and 60, respectively. For the class (MQP_2), this time is larger, with an average of 4705.2 seconds, *ac* seconds and *ac* seconds, for instances of size 40, 50, and 60, respectively. Observe that, as every feasible dual of (SDP) provides α, and β that make convex the objective function of the reformulated problem [4], and because the SDP solvers often provide dual feasible solutions as they progress, the solution of semidefinite programs can be stopped after a fixed time. This possibility is interesting for large instances since SDP solvers generally find a good solution very quickly.

5. Conclusion

In this paper, we present a computational study of `MIQCR`, a convex reformulation of general mixed-integer programs. The approach has two phases: the first phase consists of building a convex reformulation of (QP), and in the second phase the reformulated problem is submitted to a MIQP solver. Computational experiments on two types of problems show that after the reformulation by the `MIQCR` approach, most of the instances proposed can be solved within 1 hour of CPU time. However, the time to compute the parameters of the reformulation is still weighty, and an important future direction for research consists of trying to decrease the SDP solution time.

Bibliography

[1] Borchers B. A c library for semidefinite programming. *Optimization Methods and Software*, 11(1):613–623, 1999.

[2] ILOG. Ilog cplex 11.0 reference manual. *ILOG CPLEX Division, Gentilly*, 2008.

[3] Garey M.R. Johnson, D.S. Computers and intractability: A guide to the theory of np-completness. *W.H. Freeman, San Francisco*, 1979.

[4] Billionnet A. Elloumi S. Lambert, A. Solution of general mixed-integer quadratic programs through convex reformulation. *Mathematical Programming, accepted for publication*.

[5] Billionnet A. Elloumi S. Lambert, A. Linear reformulations of integer quadratic programs. *MCO 2008, september 8-10*, pages 43–51, 2008.

[6] Bonami P. Biegler L. Conn A. Cornuéjols G. Grossmann I. Laird C. Lee J. Lodi A. Margot F. Sawaya, A. An algorithmic framework for convex mixed integer nonlinear programming. *Discrete Optimization*, 5:186–204, 2005.

Billionnet Alain, Elloumi Sourour, Lambert Amélie
CEDRIC-ENSIIE
1, square de la résistance
Evry, 91026 Cedex, France
alain.billionnet@ensiie.fr
sourour.elloumi@ensiie.fr
amelie.lambert@ensiie.fr