Programmation Quadratique :
le cas non séparable – the non separable case

Dominique Quadri
Université Paris Saclay

Outline

1. QMKP non séparable
   a) Formulation mathématique
   b) Application : gestion de porte feuilles
2. Résolution
   a) Décomposition de Gauss
   b) Une linéarisation pour calculer une borne
   c) Solution exacte
3. Expérimentations

1. Quadratic Multi Knapsack Problem: mathematical formulation

$$(QMKP)\quad \left\{ \begin{array}{l} \text{maximize } f(x) = c^t x - \frac{1}{2} x^t Q x \\ s.t. \left| \begin{array}{l} Ax \leq b \\ 0 \leq x \leq u \quad x \text{ integer} \end{array} \right. \end{array} \right\}$$

- Q :matrice $n \times n$, semi définie positive definite,avec coefficients entiers
- Si Q est **diagonale**, le problème est **separable**

- Comment résoudre ce problème? :
  - En utilisant un solver commercial (cplex, par exemple): possible si Q est SDP sinon algorithms pour le non convexe et solvers comme BARON, COUENNE, BONMIN

  - Différentes méthodes avec Q diagonale (pb séparable) existent [QST07], [Mathur et al., 83], [Djerdjour et al. 88] [Bretthauer et al. 95]

Mathematical formulation (2)

- <u>An example</u>

$$(QMKP)\quad \begin{cases} \text{maximize } f(x) = 69x_1 + 71x_2 - (15x_1^2 + 2x_1x_2 + 17x_2^2) \\ s.t. \begin{cases} 81x_1 + 50x_2 \le 61 \\ 17x_1 + 2x_2 \le 105 \\ 0 \le x_1 \le 3 \quad x_1 \text{ integer} \\ 0 \le x_2 \le 2 \quad x_2 \text{ integer} \end{cases} \end{cases}$$

Application to portfolio management

- A customer must choose among *n* possible investments (asset *i*) on *m* periods (*j*). He must decide the quantity of each asset that will be placed into his portfolio.
- The buying of an asset must be done on *m* periods. We know the budget b_j devoted to each period and the estimated cost a_{ij} of asset *i* in period *j*

$$\begin{cases} \max f(x) = \langle R \mid x - B \rangle - \lambda^t (x-B)^t W (x-B) \\ s.t. \left| \begin{array}{l} Ax \leq b \\ 0 \leq x \leq u \quad x \text{ integer} \end{array} \right. \end{cases}$$

- *B*: Benchmark allocation (fixed by the market). If the customer wants to buy some assets *i*, he has to buy a minimal quantity B_i of them;
- *W*: variance-covariance matrix ;
- *R*: expected return vector;
- λ : risk-aversion (or risk-interest) coefficient;

2. Solving the separable case by transforming *QMKP* into an equivalent separable problem

- The transformed problem has to remain **convex**
- Adding constraints and/or variables is possible
- We have to end up in a separable multi-knapsack with integer or mixed variables
- A first (natural) idea: **diagonalizing** Q

$$Q = PDP^{-1}$$

  - P : transformation matrix
  - D : diagonal matrix

Transforming *QMKP* into an equivalent separable problem (2): Gauss decomposition

- First attemp : using the eigenvalues of Q
- Another way to diagonalize *Q*: Gauss decomposition

  - We make *Q* a **lower triangular matrix** using a Gauss elimination

$$T=(R^t)^{-1}Q$$

(we pre-multiply *Q* by a suitable matrix)

  - Then we again apply the Gauss elimination to make the matrix **upper triangular**

$$D=TR^{-1}=(R^t)^{-1}QR^{-1}$$

(we multiply *Q* by another suitable matrix)

  - This yields a diagonal matrix.

Transforming *QMKP* into an equivalent separable problem (3)

- R is "easy" (polynomial) to compute

$$R = \left(E_{n-1}^{t}\right)^{-1}\left(E_{n-2}^{t}\right)^{-1}\ldots\left(E_{2}^{t}\right)^{-1}\left(E_{1}^{t}\right)^{-1}$$

- E_i matrices are called Gauss elimination matrices

- They have nice properties:

  - E_i is diagonal, except for one column

  - E_i^{-1} can be easily calculated, without explicitely inverting E_i

  - There exists an iterative procedure (n-1 iterations) to compute the successive E_i

  - The elements of E_i are rational numbers (if Q is integer), issued from the successive divisions by the pivots

Transforming *QMKP* into an equivalent separable problem (4)

- Calculating the elimination matrices

$$Q = \begin{pmatrix} q_{11} & \cdots & q_{1n} \\ \vdots & & \vdots \\ q_{n1} & \cdots & q_{nn} \end{pmatrix}$$

$$E_1 = \begin{pmatrix} 1 & & & 0 \\ -\frac{q_{21}}{q_{11}} & 1 & & \\ \vdots & & \ddots & \\ -\frac{q_{n1}}{q_{11}} & 0 & & 1 \end{pmatrix}$$

$$E_1 Q E_1^t = \begin{pmatrix} q_{11} & 0 & \cdots & 0 \\ 0 & q'_{22} & \cdots & q'_{2n} \\ \vdots & \vdots & & \vdots \\ 0 & q'_{n2} & \cdots & q'_{nn} \end{pmatrix}$$

And we iterate...

A discrete separable formulation

- $Q = R^t D R$
- $x^t Q x = x^t R^t D\, R x = y^t D y$, with $y = Rx$
- $y = Rx$ iff $x = R^{-1} y$

$$(QMKP_y)\ \begin{cases} \text{maximize } g(y) = c^t R^{-1} y - \frac{1}{2} y^t D y \\ s.t. \left| \begin{array}{l} A R^{-1} y \leq b \\ y \in Y \end{array} \right. \end{cases}$$

- Y : finite set of values
- Did we progress?
- D is composed with the Gauss pivots and the elements of R are rational: less rounding errors, but there are still discrete variables…

A discrete separable formulation (2)

$$(QMKP_y)\left\{\begin{array}{l} \text{maximize } g(y) = 69y_1 + \frac{996}{15}y_2 - 15y_1^2 - \frac{254}{15}y_2^2 \\ s.t. \left| \begin{array}{l} 81y_1 + \frac{669}{15}y_2 \le 61 \\ 17y_1 + \frac{13}{15}y_2 \le 105 \\ y_1 \in Y_1 \\ y_2 \in Y_2 \end{array}\right. \end{array}\right.$$

$$y_1 \in \left\{0, \frac{1}{15}, \frac{2}{15}, 1, \frac{16}{15}, \frac{17}{15}, 2, \frac{31}{15}, \frac{32}{15}, 3, \frac{46}{15}, \frac{47}{15}\right\}$$

$$y_2 \in \{0, 1, 2\}$$

A mixed integer separable formulation (partial change of variables)

$$(QMKP_{xy}) \begin{cases} \text{maximize } g(y) = c^t R^{-1} y - \frac{1}{2} y^t D y \\ \text{s.t.} \left| \begin{array}{l} Ax \le b \\ Rx = y \\ 0 \le x \le u, \quad x \text{ integer} \end{array} \right. \end{cases}$$

- **<u>Advantages</u>:**
  - No Y set to handle
  - No irrational values

A mixed integer separable formulation (2)

$$(QMKP_{xy})\left\{\begin{array}{l} \text{maximize } g(y) = 69y_1 + \frac{996}{15}y_2 - 15y_1^2 - \frac{254}{15}y_2^2 \\ s.t. \begin{cases} 81x_1 + 50x_2 \leq 61 \\ 17x_1 + 2x_2 \leq 105 \\ x_1 + \frac{1}{15}x_2 = y_1 \\ x_2 = y_2 \\ x_1 \in \{0,1,2,3\} \\ x_2 \in \{0,1,2\} \end{cases} \end{array}\right.$$

A linearization to compute an upper bound

$$(QMKP_{xy})\ \begin{cases} \text{maximize } g(y) = c'^t y - \frac{1}{2} y^t D y \\ s.t. \left| \begin{array}{c} Ax \le b \\ Rx = y \\ 0 \le x \le u, \quad x \text{ integer} \end{array} \right. \end{cases} \quad \text{with } c' = c^t R^{-1}$$

- The objective function of $(QMKP_{xy})$: $\sum g_i(y_i)$

  where

$$g_i(y_i) = c_i' y_i - \frac{1}{2} d_i y_i^2$$

A linearization to compute an upper bound (2)

- We first compute lower and upper bounds B_i^- and B_i^+ for each variable y_i:

$$\forall i \in \{1,\ldots,n\}\ B_i^- \leq y_i \leq B_i^+ \quad \text{with} \quad \begin{cases} B_i^- = \sum R_{ij} u_j \\ B_i^+ = \sum R_{ij} u_j \end{cases}$$

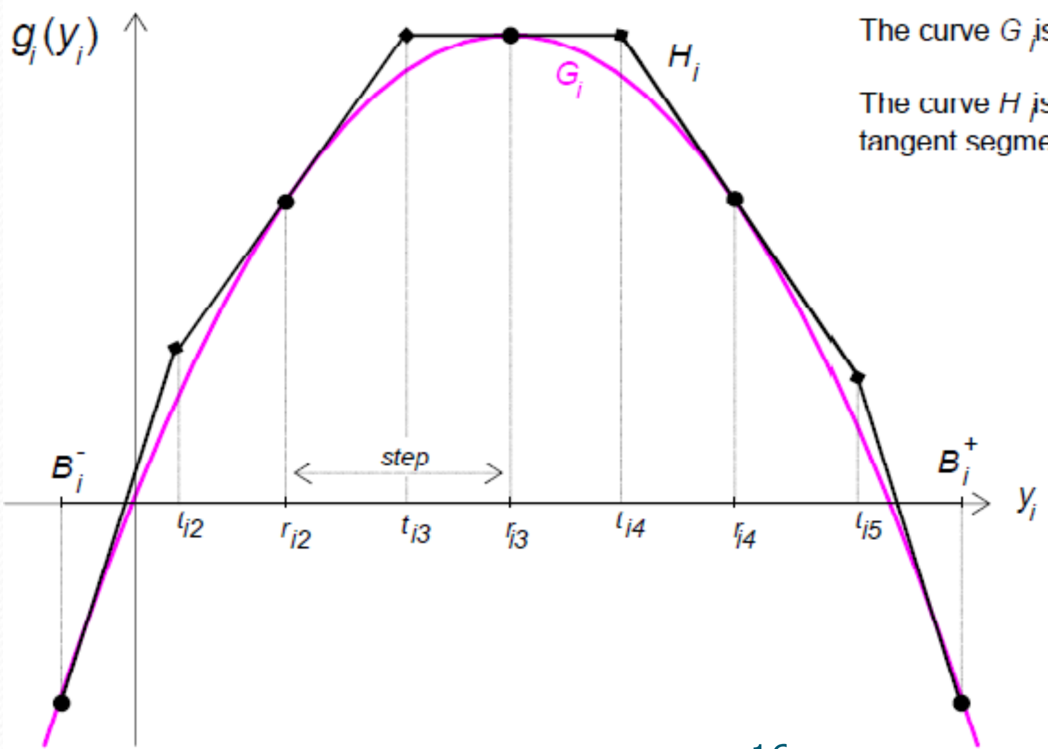
- Improvement of theses bounds on variables by pre-processing

$$(QMKP_{xy}) \begin{cases} \text{maximize } g(y) = c'^t y - \frac{1}{2} y^t D y \\ \quad Ax \leq b \\ s.t. \quad Rx = y \\ \quad 0 \leq x \leq u, \quad x \text{ integer} \end{cases}$$

A linearization to compute an upper bound (3)

- We then overestimate each quadratic sub-function g_i with a piecewise linear one, h_i :

$$g(y) = \sum g_i(y_i) \leq \sum h_i(y_i)$$



The curve G_i is the parabola.

The curve H_i is the linear piecewise curve formed by the K=5 tangent segments

$$g_i(y_i) = c_i' y_i - \frac{1}{2} d_i y_i^2$$

A linearization to compute an upper bound (4)

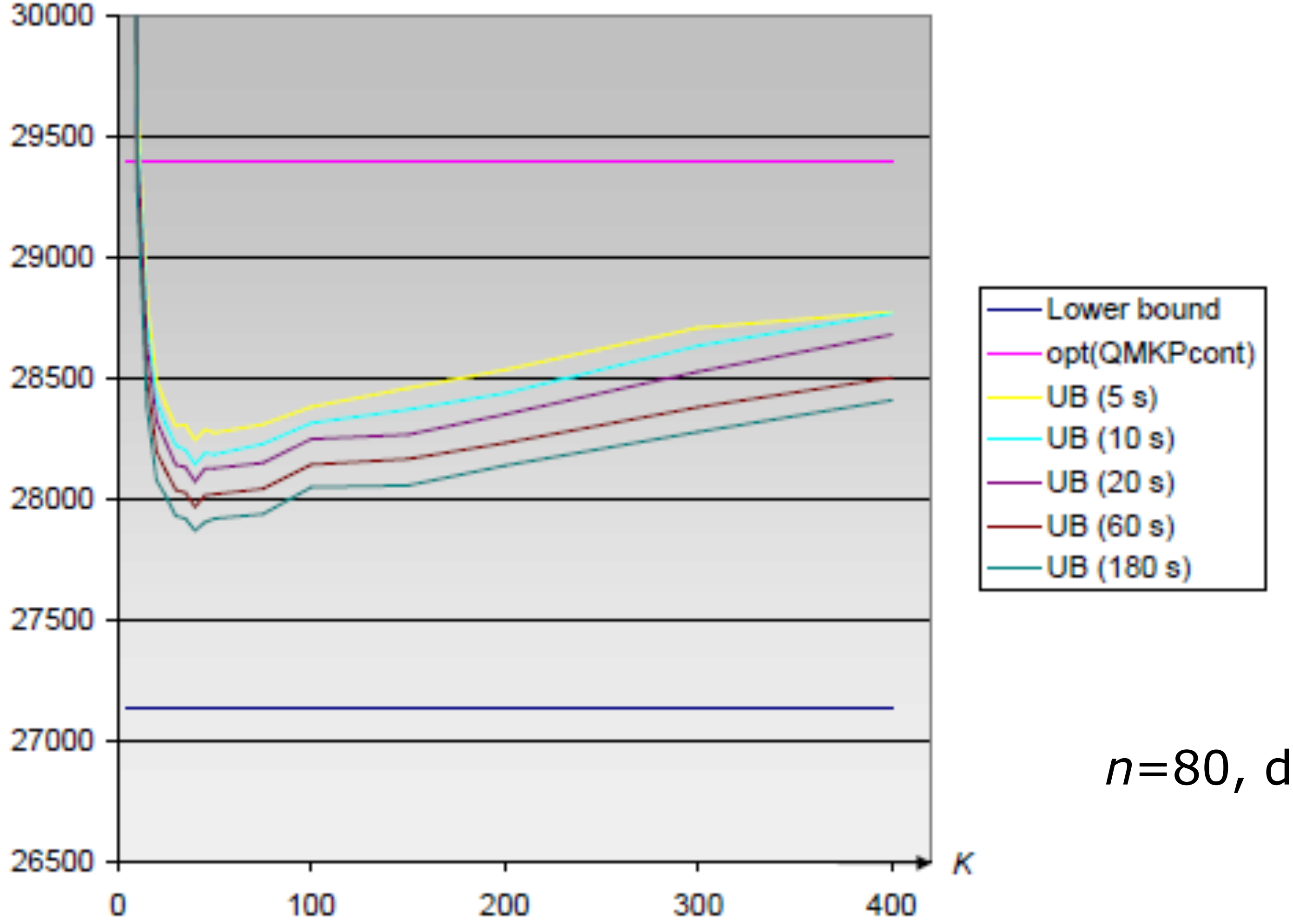
- We now solve the problem obtained by substituting g_i with h_i :
- K is the number of segments tangent to the curve;
- y_i is replaced with K continuous z_{ik} variables by stating:

$$y_i = B_i^- + \sum_{k=1}^{K} z_{ik}$$

$$(LP_{x,z}) \left\{ \begin{array}{ll} \max \ h(z) = cst + \sum \sum \alpha_{ik} z_{ik} & \\ s.t. \left| \begin{array}{ll} Ax \leq b & \\ \sum R_{ij} x_j = B_i^- + \sum z_{ik} & i = 1,\ldots,n \\ 0 \leq x_i \leq u_i, x_i \text{ integer} & i = 1,\ldots,n \\ 0 \leq z_{ik} \leq Z_{ik} & i = 1,\ldots,n, k = 1,\ldots,K \end{array} \right. \end{array} \right\} \Rightarrow UB_K$$

- where α_{ik} is the slope of the k^{th} segment;

Influence of the parameter K on the bound UB_K



n=80, d=30

An exact method

- Using the bound UB_K in a branch-and-bound algorithm?
- $$UB_K = opt(QMKP_{xy}) = opt(QMKP)$$
- Computation of a good lower bound
  - A heuristic method
  - Reconstruction of a solution
- Intensification around the two best known integer solutions

Numerical experiments

- Correlated instances derived from [Chu and Beasley 98]

- Generating a symmetric definite positive matrix Q and adding a quadratic part to the linear function

- 3 types of instances, corresponding to 3 different densities (proportion of nonzero coefficients in the objective function).

Comparison of the quality of the upper bounds

Instances				$Z[\overline{QMKP}]$ (Cplex 10.0)	$Z[\overline{QP_{x,y}}]$ (Cplex 10.0)	$Z[LP_{x,z}]$ Our approach		
n	m	α	K	Gap (%)	Gap (%)	Gap (%)	% Opt	♯ var
10	10	0.25	100	0.012	0.012	0.0	100	1010
10	10	0.5	100	0.014	0.014	0.0	100	1010
10	10	0.75	100	0.017	0.017	0.0	100	1010
20	20	0.25	200	0.022	0.022	0.0	100	4020
20	20	0.5	200	0.023	0.023	0.0	100	4020
20	20	0.75	200	0.031	0.031	0.0	100	4020
30	30	0.25	300	0.027	0.027	0.0	100	9030
30	30	0.5	300	0.032	0.032	0.0	100	9030
30	30	0.75	300	0.033	0.033	0.0	100	9030
40	30	0.25	400	0.041	0.041	0.0	100	16040
40	30	0.5	400	0.052	0.052	0.0	100	16040
40	30	0.75	400	0.064	0.064	0.0	100	16040

Comparison of the CPU times

Instances				$Z[\overline{QMKP}]$ (Cplex 10.0)	$Z[\overline{QP_{x,y}}]$ (Cplex 10.0)	$Z[LP_{x,z}]$ Our approach
n	m	α	K			
10	10	0.25	100	0.002	0.001	0.015
10	10	0.5	100	0.002	0.001	0.013
10	10	0.75	100	0.002	0.001	0.016
20	20	0.25	200	0.002	0.001	0.148
20	20	0.5	200	0.003	0.003	0.1412
20	20	0.75	200	0.091	0.098	0.12
30	30	0.25	300	0.13	0.14	0.24
30	30	0.5	300	0.27	0.44	1.33
30	30	0.75	300	0.17	0.34	0.97
40	30	0.25	400	4.21	6.96	7.89
40	30	0.5	400	4.51	7.16	8.12
40	30	0.75	400	4.96	7.78	8.53

Comparison of the CPU times to exactly solve (QMKP)

n	Density	Time (s) opt QMKPxy (sep pb) using Cplex 12	Time (s) opt QMKP (initial nonsep pb) using Cplex 12	Time opt asympt lin including preprocessing & lower bound (s)	Improvement
40	25%	2,71	1,83	**4,45**	time mult. by 2,4
	50%	3,07	1,92	**2,54**	time mult. by 1,3
	75%	0,86	0,57	**1,73**	time mult. by 3
50	25%	40,39	22,68	**8,89**	time div. by 2,6
	50%	10,84	6,10	**4,90**	time div. by 1,2
	75%	9,97	6,12	**3,76**	time div. by 1,6
60	25%	1454,16	896,94	**484,29**	time div. by 1,9
	50%	304,19	2 077,26	**15,79**	time div. by 131,6
	75%	231,98	1 161,23	**13,40**	time div. by 86,7

Each line represents the average results on 10 instances solved to optimality – m=30

Adjustment of the parameters and characteristics of the asymptotic linearization

n	Density	K	Intens. 1	Intens. 2	gap asympt lin	Time opt (including preprocessing & lower bound) (seconds)	Time asympt. Lin. (s)	Time lower bound (s)	Time pre-processing (s)	# attempts
	25%	5	10	0	0,001%	4,45	0,47	0,49	3,49	1
40	50%	5	10	0	0,001%	2,54	0,37	0,16	2,01	1
	75%	5,5	12	0	0,001%	1,73	0,22	0,12	1,38	1,1
	25%	5	16	0	0,001%	8,89	5,36	0,58	2,95	1
50	50%	5	32,8	0	0,001%	4,90	2,21	0,17	2,51	1,2
	75%	5	21,9	0	0,001%	3,76	1,52	0,18	2,06	1,1
	25%	10	33	4	0,002%	484,29	475,25	1,67	7,37	1,1

Conclusion and future works

- Finding the good adjustment of the parameters remains difficult
- Future works: solving exactly larger instances
- Dealing with non-convex problems