Reformulation convexe des programmes quadratiques entiers : un algorithme de Branch and Bound fondé sur la structure du problème reformulé

Alain Billionnet[1], Sourour Elloumi[1], Amélie Lambert[1]

CEDRIC-ENSIIE, 1 square de la résistance, F-91025 Evry cedex, France
{alain.billionnet,sourour.elloumi,amelie.lambert}@ensiie.fr

Mots-Clés : *programmation quadratique entière, reformulation convexe, Branch and Bound*

Une reformulation quadratique convexe : `IQCR` (`Integer Quadratic Convex Reformulation`)

De nombreux problèmes de recherche opérationnelle peuvent se modéliser sous la forme d'un programme mathématique avec variables entières dont la fonction objectif est quadratique quelconque et est soumise à des contraintes linéaires. Un problème de ce type peut s'écrire sous la forme :

$$(QP) \left\{ \begin{array}{ll} Min & f(x) = x^T Q x + c^T x \\ s.c & Ax = b \quad Dx \leq e \quad 0 \leq x \leq u \quad x \in \mathbb{N}^n \end{array} \right.$$

La fonction f n'étant pas convexe, même la relaxation continue de (QP) est difficile à résoudre. La reformulation `IQCR` [1] contourne cette difficulté en trouvant une reformulation équivalente de (QP) par un nouveau programme quadratique $(QP_{\alpha,\beta})$, équivalent à (QP). La fonction objectif de $(QP_{\alpha,\beta})$ s'écrit ainsi :

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

où les variables y sont des variables additionnelles qui seront contraintes de vérifier $y_{ij} = x_i x_j$. Nous forçons cette égalité par la linéarisation utilisée dans [1] et [2] qui ajoute d'autres variables et des contraintes, toutes linéaires. Nous nous limitons aux paramètres α et β qui rendent $f_{\alpha,\beta}(x,y)$ convexe. Le programme $(QP_{\alpha,\beta})$ ainsi obtenu est quadratique convexe en nombres entiers et sa résolution fournit une solution optimale pour (QP). De plus, parmi toutes les valeurs possibles de α et β, nous nous intéressons à celles qui maximisent la borne obtenue par relaxation continue et nous montrons qu'il est possible de les calculer par la résolution d'une relaxation semi-définie de (QP).

Grâce à la convexité de sa fonction objectif, le programme reformulé $(QP_{\alpha,\beta})$ peut être résolu par un solveur standard [1]. Or, dans ce travail, nous profitons de la structure de ce programme pour essayer d'aller plus vite que le solveur standard. Comme évoqué ci-dessus, la reformulation en $(QP_{\alpha,\beta})$ ajoute un nombre important de contraintes et de variables. Cependant, la relaxation continue de $(QP_{\alpha,\beta})$, que nous notons $(\overline{QP}_{\alpha,\beta})$, possède une propriété de projection intéressante. En effet, la projection de $(\overline{QP}_{\alpha,\beta})$ sur les variables x et y, donne le programme $(P_{\alpha,\beta})$ suivant :

$$(P_{\alpha,\beta}) \begin{cases} Min & f_{\alpha,\beta}(x,y) & \\ s.c. & Ax = b & \\ & Dx \leq e & \\ & y_{ij} = y_{ji} & \forall i, \forall j \\ & y_{ij} \leq u_j x_i & \forall i, \forall j \\ & y_{ij} \leq u_i x_j & \forall i, \forall j \\ & y_{ij} \geq x_i u_j + x_j u_i - u_i u_j & \forall i, \forall j \\ & y_{ij} \geq 0 & \forall i, \forall j \end{cases}$$

dont la valeur optimale est égale à la valeur optimale de $(\overline{QP}_{\alpha,\beta})$. Seulement, la taille de $(P_{\alpha,\beta})$, qui est en $O(n^2)$, est significativement plus petite que celle de $(\overline{QP}_{\alpha,\beta})$ [1].

Un algorithme spécifique de Branch and Bound

Nous proposons un algorithme de Branch and Bound, basé sur cette propriété de projection, qui permet de trouver la solution optimale de de $(QP_{\alpha,\beta})$ en résolvant à chaque noeud de l'arbre un problème de plus petite taille. En notant $(P^e_{\alpha,\beta})$, le problème $(P_{\alpha,\beta})$ avec la contrainte $x \in \mathbb{N}^n$, nous avons la propriété suivante :

Proposition 1 *Une solution optimale de* $(QP_{\alpha,\beta})$ *peut être trouvée en résolvant le programme* $(P^e_{\alpha,\beta})$ *augmenté des contraintes* $x_i x_i = y_{ij}$

Ainsi, une manière de trouver une solution optimale de $(QP_{\alpha,\beta})$ est de résoudre à chaque noeud de l'arbre de résolution le problème $(P^e_{\alpha,\beta})$, puis de brancher de la manière suivante :
Soit $(\bar{x}, \bar{y})$ une solution du noeud courant, plusieurs cas de figure se présentent :

1. Si $(\bar{x}, \bar{y})$ respecte la propriété $\forall i, \forall j, \bar{x}_i\bar{x}_j = \bar{y}_{ij}$, alors $\bar{x}$ est une solution réalisable de (QP) et $f(\bar{x}) = f_{\alpha,\beta}(\bar{x}, \bar{y})$
2. Si $\exists i, j$ tels que $\bar{x}_i\bar{x}_j \neq \bar{y}_{ij}$, nous branchons de la façon suivante :
   a. La branche où $x_i \neq \bar{x}_i$. Cette branche donne naissance à deux sous arbres, dans le premier on ajoute à la formulation $(P^e_{\alpha,\beta})$ la contrainte $x_i \leq \bar{x}_i - 1$ et dans le deuxième la contrainte $x_i \geq \bar{x}_i + 1$.
   b. La branche où $x_i = \bar{x}_i$ et $x_j \neq \bar{x}_j$. Cette branche donne naissance à deux sous arbres, dans le premier on ajoute à la formulation $(P^e_{\alpha,\beta})$ les contraintes $x_i = \bar{x}_i$ et $x_j \leq \bar{x}_j - 1$ et dans le deuxième les contraintes $x_i = \bar{x}_i$ et $x_j \geq \bar{x}_j + 1$.
   c. La branche où $x_i = \bar{x}_i$, $x_j = \bar{x}_j$ et $y_{ij} = \bar{x}_i\bar{x}_j$. Dans le sous arbre, on ajoute à la formulation $(P^e_{\alpha,\beta})$ les contraintes $x_i = \bar{x}_i$, $x_j = \bar{x}_j$ et $y_{ij} = \bar{x}_i\bar{x}_j$.

Expérimentations

Nous avons testé cet algorithme sur des instances de programmes quadratiques non convexes soumis à une contrainte d'égalité qui ont été générées aléatoirement et déjà utilisées dans [1] et [2]. Malgré une implémentation naïve, cet algorithme s'avère compétitif avec la résolution directe de $(QP_{\alpha,\beta})$ par un solveur standard. Le temps moyen de résolution est divisé par 2 par rapport à la résolution par Cplex.

Références

[1] Billionnet, A., Elloumi, S., Lambert, A. : Extending the QCR method to the case of general mixed-integer program. Mathematical Programming. Accepté pour publication (2009)

[2] Billionnet, A., Elloumi, S., Lambert, A. : Linear Reformulations of Integer Quadratic Programs. MCO 2008, september 8-10, 43-51 (2008)