Programmation quadratique en variables 0-1 avec contraintes linéaires

A. Billionnet[1], S. Elloumi[2], et M.-C. Plateau[2]

[1] CEDRIC, Conservatoire National des Arts et Métiers, 292 rue Saint-Martin, 75141 Paris
{mc.plateau,elloumi}@cnam.fr
[2] CEDRIC, Institut d'Informatique d'Entreprise, 18 allée Jean Rostand, 91025 Evry
billionnet@iie.cnam.fr

Nous considérons le problème suivant de programmation quadratique en variables 0-1 sous des contraintes linéaires :

$$(QP) \quad : \quad \text{Min } \{q(x) = x^tQx + c^tx : Ax = b, A'x \leq b', \ x \in \{0,1\}^n\}$$

où Q est une matrice symétrique d'ordre n, A (resp. A') est une matrice $m \times n$ (resp. $p \times n$) et c (resp. b et b') est un vecteur de n (resp. m et p) composantes.

Le modèle (QP) est très général et permet de formuler de nombreux problèmes d'Optimisation Combinatoire parmi lesquels nous pouvons citer l'affectation quadratique, certains problèmes de placement de tâches ou d'investissement, et plusieurs problèmes d'optimisation dans les graphes.

Nous nous intéressons à la résolution exacte du problème (QP) par la méthode suivante en deux phases. La Phase I, qui peut être vue comme une phase de prétraitement, consiste à transformer (QP) en un problème (QP_{α^*,u^*}) avec $\alpha^* \in R^{n\times m}$ et $u^* \in R^n$:

$$(QP_{\alpha^*,u^*}) \quad : \quad \text{Min } \{q_{\alpha^*,u^*}(x) : Ax = b, \ A'x \leq b', \ x \in \{0,1\}^n\}$$

où $\forall \alpha \in R^{n\times m}$ et $u \in R^n$: $q_{\alpha,u}(x) = q(x) + \sum\limits_{i=1}^{n}\sum\limits_{k=1}^{m} \alpha_{ki}x_i \left(\sum\limits_{j=1}^{n} a_{kj}x_j - b_k\right) + \sum_{i=1}^n u_i(x_i^2 - x_i)$

Le problème (QP_{α^*,u^*}) possède les trois propriétés suivantes : (1) il est équivalent à (QP), (2) la résolution de sa relaxation continue est un problème d'optimisation quadratique convexe, (3) la valeur optimale de cette relaxation continue est la meilleure possible dans le sens suivant :

$$\begin{aligned} &\text{Min } \{q_{\alpha^*,u^*}(x) : Ax = b, \ A'x \leq b', \ x \in [0,1]^n\} \\ = &\max_{\alpha,u: q_{\alpha,u} convexe} \text{Min } \{q_{\alpha,u}(x) : Ax = b, \ A'x \leq b', \ x \in [0,1]^n\} \end{aligned}$$

La Phase II consiste simplement à soumettre (QP_{α^*,u^*}) à un solveur de programmes quadratiques convexes en nombres entiers. Nous montrons que α^* et u^* peuvent être déterminés en résolvant le programme semidefini suivant :

$$(SDQP) \begin{cases} \text{Min } c^tx + \sum\limits_{i=1}^{n}\sum\limits_{j=1}^{n} Q_{ij}X_{ij} \\ \text{s.t.} \quad X_{ii} = x_i \qquad i = 1, \ldots, n \qquad (1) \\ \qquad -b_kx_i + \sum\limits_{j=1}^{n} a_{kj}X_{ij} = 0 \qquad i = 1, \ldots, n; \ k = 1, \ldots, m \qquad (2) \\ \qquad Ax = b \\ \qquad A'x \leq b' \\ \qquad \begin{pmatrix} 1 & x^t \\ x & X \end{pmatrix} \succeq 0 \\ \qquad x \in R^n, \ X \in R^{n\times n} \end{cases}$$

Les composantes de α^* (respectivement u^*) sont données par les valeurs des variables duales associées à la contrainte (2) (respectivement (1)) à l'optimum de $(SDQP)$.

Afin d'évaluer cette nouvelle méthode générale de résolution des programmes quadratiques en 0-1, nous l'avons utilisée pour résoudre différents problèmes d'optimisation combinatoire et notamment le problème du k-sous-graphe le plus dense et le problème de la bipartition d'un graphe. Notre implémentation s'appuie sur le logiciels SB [2] pour la première phase de la méthode, c'est-à-dire la résolution de $(SDQP)$ et sur CPLEX 9 pour la deuxième phase, c'est-à-dire pour la résolution de (QP_{α^*,u^*}).

Nous constatons, pour de nombreuses instances, une nette supériorité de notre approche générale par rapport à d'autres méthodes de résolution exacte dont certaines sont spécifiques au problème résolu. Quelle que soit la densité des graphes considérés, la méthode permet de résoudre le problème du k-sous-graphe le plus dense avec 80 sommets (quelle que soit la valeur de k) et le problème de l'équipartition avec 90 sommets.

Références

1. Billionnet,A. and Elloumi,S. and Plateau,M.C. : Convex quadratic programming for exact solution of 0-1 quadratic programs. Technical Report CEDRIC. http ://cedric.cnam.fr/PUBLIS/RC.pdf (2005)
2. Helmberg, C. : A C++ implementation of the Spectral Bundle Method. Manual version 1.1.1, http ://www.zib.de/helmberg/SBmethod (2000)
3. Pisinger, D. : Exact solution of p-dispersion problems. Technical Report DIKU 99/14, University of Copenhagen (1999)
4. Hammer, P.L., A. Rubin : Some remarks on quadratic programming with 0-1 variables. RAIRO 3, pp.67-79 (1970)
5. Karisch,S.E., Rendl,F., Clausen,J. : Solving Graph Bisection Problems with Semidefinite Programming. INFORMS Journal on Computing, pp.177-191,12 :3 (2000)