Decomposição Lagrangeana com Geração de Colunas para o Problema de Programação Quadrática Binária Irrestrita

Geraldo Regis Mauri[1,2], Luiz Antonio Nogueira Lorena[2]

[1]Centro de Ciências Agrárias, Departamento de Engenharia Rural
Universidade Federal do Espírito Santo - UFES
[2]Laboratório Associado de Computação e Matemática Aplicada
Instituto Nacional de Pesquisas Espaciais - INPE
mauri@cca.ufes.br, {mauri,lorena}@lac.inpe.br

Abstract. *The Unconstrained Binary Quadratic Programming Problem (PQ) consists of optimizing a quadratic objective by suitable choice of binary decisions variables. This paper proposes a new alternative of lagrangean decomposition based on column generation technique to solve PQ. The results show the efficiency of the proposed method over traditional lagrangean relaxations and other methods found in the literature.*

Resumo. *O Problema de Programação Quadrática Binária Irrestrita (PQ) consiste em otimizar uma função quadrática por meio da escolha de valores binários apropriados para as variáveis de decisão. Este trabalho propõe uma nova alternativa de decomposição lagrangeana baseada na técnica de geração de colunas para resolução do PQ. Os resultados mostram a eficiência do método proposto em relação a métodos tradicionais de relaxação lagrangeana e outros métodos encontrados na literatura.*

1. Introdução

O Problema de Programação Quadrática Binária Irrestrita (*PQ*) é um dos problemas clássicos em otimização não-linear que consiste em maximizar (ou minimizar) uma função objetivo quadrática por meio da escolha de valores apropriados para as variáveis de decisão binárias [Beasley 1998]. O *PQ* é um problema *NP-Hard* [Billionnet e Elloumi 2007] e apresenta uma grande quantidade de aplicações em diversas áreas, como por exemplo: biologia molecular; planejamento de investimentos e análise financeira; e alguns problemas do tipo CAD (*Computer Aided Design*). Além disso, o *PQ* ainda aborda inúmeros problemas modelados por meio de grafos, como clique máximo, máximo conjunto independente, e outros.

Este trabalho propõe uma nova alternativa de decomposição lagrangeana que utiliza a técnica de geração de colunas para encontrar soluções e limitantes para o *PQ*. Esse método trata um modelo linear inteiro misto do *PQ* que tem restrições representadas por meio de um grafo que é particionado com o uso de uma biblioteca de heurísticas denominada METIS [Karypis e Kumar 1998]. O problema original é decomposto formando um problema coordenador (problema mestre restrito) e um subproblema para cada *cluster*, que são utilizados para geração de novas colunas para o problema coordenador.

O restante do artigo está organizado como segue. A Seção 2 apresenta o método proposto, e alguns métodos alternativos para obtenção de limitantes para o *PQ* são descritos na Seção 3. Os resultados computacionais são apresentados na Seção 4, e as conclusões são resumidas na Seção 5.

2. Decomposição lagrangeana com geração de colunas

A decomposição lagrangeana é um caso especial de relaxação lagrangeana que consiste em dividir o problema original em vários subproblemas e criar uma cópia das variáveis de decisão em cada um dos subproblemas gerados. Essas "cópias" são utilizadas nas restrições dos subproblemas, e restrições que garantem a igualdade entre as variáveis originais e suas cópias são relaxadas no sentido lagrangeano [Chardaire e Sutter 1995].

Considerando uma matriz de números reais $\mathbf{Q} = [q_{ij}]_{mxm}$, o *PQ* pode ser formulado pela expressão (1).

$$PQ: \quad v(PQ) = \max_{x\ binário} \sum_{i=1}^{m}\sum_{j=1}^{m} q_{ij} x_i x_j \tag{1}$$

O *PQ* pode ser linearizado por meio da substituição dos termos quadráticos $x_i x_j$ pela variável contínua w_{ij} e por restrições que garantam que $w_{ij} = x_i x_j$. Logo, tem-se uma versão linear inteira mista de *PQ* (2-7). Por convenção, esse modelo será chamado *PQL*.

$$PQL: \quad v(PQL) = \mathbf{Max} \quad \sum_{i=1}^{m} q_{ii} x_i + \sum_{i \neq j} q_{ij} w_{ij} \tag{2}$$

Sujeito a

$$w_{ij} - x_i \leq 0 \qquad i \neq j, q_{ij} > 0 \tag{3}$$

$$w_{ij} - x_j \leq 0 \qquad i \neq j, q_{ij} > 0 \tag{4}$$

$$x_i + x_j - w_{ij} \leq 1 \qquad i \neq j, q_{ij} < 0 \tag{5}$$

$$w_{ij} \geq 0 \qquad i \neq j, q_{ij} < 0 \tag{6}$$

$$x_i \in \{0,1\} \qquad i = 1,...,m \tag{7}$$

A partir da matriz **Q**, pode-se criar um grafo $G=(V,E)$ com $V = \{1,...,m\}$ e uma matriz de adjacências $\mathbf{E} = [e_{ij}]_{mxm}$, $e_{ij} = 1$ se $q_{ij} \neq 0$ e $e_{ij} = 0$ se $q_{ij} = 0$. Particionando o grafo G em n ($n \leq m$) *clusters* independentes, tem-se $V = V_1 \cup V_2 \cup ... \cup V_n$, onde $V_i \cap V_j = \varnothing$, $\forall\ i \neq j$, $G_i = (V_i, A_i)$, $i = 1,...,n$, e $X_i = V - V_i$, $i = 1,...,n$. Logo, tem-se:

$$PQL^n: \ v(PQL^n) = \mathbf{Max} \sum_{k=1}^{n} \left(\sum_{i \in V_k} q_{ii} x_i + \sum_{i \in V_k; j \in V_k; i \neq j} q_{ij} w_{ij} + \sum_{i \in V_k; j \in X_k; i \neq j} q_{ij} w_{ij'} \right) \tag{8}$$

Sujeito a

$$w_{ij} \leq x_i \qquad i \in V_k, j \in V_k, i \neq j, q_{ij} > 0, k = 1,...,n \tag{9}$$

$$w_{ij} \leq x_j \qquad i \in V_k, j \in V_k, i \neq j, q_{ij} > 0, k = 1,...,n \tag{10}$$

$$x_i + x_j - w_{ij} \leq 1 \qquad i \in V_k, j \in V_k, i \neq j, q_{ij} < 0, k = 1,...,n \tag{11}$$

$$w_{ij} \geq 0 \qquad i \in V_k, j \in V_k, i \neq j, q_{ij} < 0, k = 1,...,n \tag{12}$$

$$w_{ij'} \leq x_i \qquad i \in V_k, j \in X_k, i \neq j, q_{ij} > 0, k = 1,...,n \tag{13}$$

$$w_{ij'} \leq x_{j'}^k \qquad i \in V_k, j \in X_k, i \neq j, q_{ij} > 0, k = 1,...,n \tag{14}$$

$$x_i + x_{j'}^k - w_{ij'} \leq 1 \qquad i \in V_k, j \in X_k, i \neq j, q_{ij} < 0, k = 1,...,n \tag{15}$$

$$w_{ij'} \geq 0 \qquad i \in V_k, j \in X_k, i \neq j, q_{ij} < 0, k = 1,...,n \tag{16}$$

$$x_j = x_{j'}^k \qquad j \in X_k, k = 1,...,n \tag{17}$$

$$w_{ij'} = w_{ji'} \qquad i \in V_k, j \in X_k, j > i, k = 1,...,n \tag{18}$$

$$x_i \in \{0,1\} \qquad i \in V_k, k = 1,...,n \tag{19}$$

Nesse modelo, as variáveis $x^k_{j'}$ representam as cópias do vértice j (j') no *cluster k*, e as variáveis $w_{ij'}$ representam as arestas entre os vértices i e j' (cópia de j). As restrições (9)-(12) tratam apenas as arestas (i,j) cujos vértices são internos ao *cluster* (sub-grafo) k. Já as restrições (13)-(16) tratam as arestas (i,j) cujos vértices estão em *clusters* distintos. As restrições (17) e (18) são as restrições de cópia, que garantem a igualdade entre as variáveis originais e suas cópias. Com intuito de facilitar a compreensão do método proposto, esse modelo (9-19) pode ser representado matricialmente da seguinte forma:

$$PQL^n\text{: } v(PQL^n) = \mathbf{Max} \sum_{k=1}^{n} \left(\mathbf{q}_x^k \mathbf{x}^k + \mathbf{0}\mathbf{x}'^k + \mathbf{q}_w^k \mathbf{w}^k + \mathbf{q}_{w'}^k \mathbf{w}'^k \right) \tag{20}$$

$$\textbf{Sujeito a} \quad \begin{bmatrix} \mathbf{A_1} & \cdots & \mathbf{A}_n \\ \mathbf{B_1} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_n \end{bmatrix} \times \left[[\mathbf{x} \quad \mathbf{x}' \quad \mathbf{w} \quad \mathbf{w}']^1 \cdots [\mathbf{x} \quad \mathbf{x}' \quad \mathbf{w} \quad \mathbf{w}']^n \right]^{\mathrm{T}} \sim \mathbf{b} \tag{21}$$

$\mathbf{q}_x^k$ é um vetor com os coeficientes das variáveis x_i presentes no *cluster k*; $\mathbf{q}_w^k$ é um vetor com os coeficientes das variáveis w_{ij} presentes no *cluster k*; $\mathbf{q}_{w'}^k$ é um vetor com os coeficientes das variáveis $w_{ij'}$ presentes no *cluster k*; $\mathbf{B_k}$ é uma matriz com os coeficientes das variáveis presentes nas restrições (9) a (16); $\mathbf{A_k}$ é uma matriz com os coeficientes das variáveis presentes restrições de cópia (17-18); $\mathbf{x}$ é um vetor com os valores das variáveis de decisão x_i; $\mathbf{x'}$ é um vetor com os valores das variáveis de decisão $x^k_{i'}$; $\mathbf{w}$ é um vetor com os valores das variáveis de decisão w_{ij}; $\mathbf{w'}$ é um vetor com os valores das variáveis de decisão $w_{ij'}$; Por fim, ~ representa os operadores relacionais (≤, ≥ ou =) e $\mathbf{b}$ os valores do lado direito das restrições (0 ou 1).

Relaxando as restrições (17) e (18) no sentido lagrangeano por meio do vetor de multiplicadores $\boldsymbol{\mu}$, ($\boldsymbol{\mu}$ irrestrito), o problema PQL^n (indiretamente o PQ) pode ser dividido em n subproblemas independentes. Considerando R_k como sendo as restrições associadas ao subproblema k, k=1,...,n, e d o número de restrições relaxadas, cada subproblema pode ser representado como descrito em (22), e a solução da decomposição do problema PQL em *n clusters* é dada pela expressão (23).

$$DL_\mu PQL_k\text{: } v(DL_\mu PQL_k) =$$

$$\max_{\mathbf{x}^k, \mathbf{x}'^k, \mathbf{w}^k, \mathbf{w}'^k \in R_k} \left\{ \left(\mathbf{q}_x^k - \boldsymbol{\mu}\mathbf{A}_k\right)\mathbf{x}^k + \left(-\boldsymbol{\mu}\mathbf{A}_k\right)\mathbf{x}'^k + \left(\mathbf{q}_w^k - \boldsymbol{\mu}\mathbf{A}_k\right)\mathbf{w}^k + \left(\mathbf{q}_{w'}^k - \boldsymbol{\mu}\mathbf{A}_k\right)\mathbf{w}'^k \right\} \tag{22}$$

$$DL_\mu PQL^n\text{: } v(DL_\mu PQL^n) = \sum_{k=1}^{n} DL_\mu PQL_k - \sum_{p=1}^{d} \mu_p \tag{23}$$

A implementação clássica da técnica de geração de colunas utiliza um problema coordenador, ou *Problema Mestre Restrito* (*PMR*), e subproblemas geradores de colunas para formá-lo. O *PMR*, por meio de suas variáveis duais, direciona os subproblemas na busca de novas colunas. Assim, aplicando a decomposição *Dantzig-Wolfe* para a relaxação linear (*PL*) do problema PQL^n tem-se:

$$PMR_{PL}\text{: } v(PMR_{PL}) = \mathbf{Max} \sum_{k=1}^{n} \boldsymbol{\lambda}_{sk} \left(\mathbf{q}_x^s \mathbf{x}^{sk} + \mathbf{0}\mathbf{x}'^{sk} + \mathbf{q}_w^s \mathbf{w}^{sk} + \mathbf{q}_{w'}^s \mathbf{w}'^{sk} \right) \tag{24}$$

Sujeito à

$$\sum_{k=1}^{n} \sum_{s \in S_k} \boldsymbol{\lambda}_{sk} \left(\mathbf{A}_k \mathbf{y}^{sk} \right) \sim \mathbf{b}, onde\ \mathbf{y} = [\mathbf{x} \quad \mathbf{x}' \quad \mathbf{w} \quad \mathbf{w}']^{\mathrm{T}} \tag{25}$$

$$\sum_{s \in S_k} \boldsymbol{\lambda}_{sk} = 1 \qquad \forall k \in \{1, ..., n\} \tag{26}$$

$$\boldsymbol{\lambda}_{sk} \geq 0 \qquad \forall k \in \{1, ..., n\}, \quad s \in S_k \tag{27}$$

O conjunto de pontos extremos de R_k associados com as colunas geradas no *PMR* é dado por S_k. β_k é a variável dual associada com a *k-ésima* restrição de convexidade de (26). $\mathbf{x}^{sk}$, $\mathbf{x'}^{sk}$, $\mathbf{w}^{sk}$ e $\mathbf{w'}^{sk}$ são vetores que definem os pontos extremos $s \in S_k$, ou seja, as soluções viáveis do subproblema definido pelo *cluster k*. λ_{sk} é a variável de decisão correspondente ao ponto extremo $s \in S_k$. Para cada subproblema k (22), pode-se substituir o vetor de multiplicadores lagrangeanos $\boldsymbol{\mu}$ pelo vetor de variáveis duais $\boldsymbol{\alpha}$ associado com as restrições (25), e de uma forma alternativa, cada subproblema pode ser descrito pela expressão (28), e a relaxação do *PQL*, em *n clusters*, por (29).

$$Z_k = \max_{\mathbf{x}^k, \mathbf{x'}^k, \mathbf{w}^k, \mathbf{w'}^k \in R_k} \left\{ \left(\mathbf{q}_x^k - \boldsymbol{\alpha}\mathbf{A}_k\right)\mathbf{x}^k + \left(-\boldsymbol{\alpha}\mathbf{A}_k\right)\mathbf{x'}^k + \left(\mathbf{q}_w^k - \boldsymbol{\alpha}\mathbf{A}_k\right)\mathbf{w}^k + \left(\mathbf{q}_{w'}^k - \boldsymbol{\alpha}\mathbf{A}_k\right)\mathbf{w'}^k \right\} \quad (28)$$

$$DL_\alpha PQL^n\text{: } v(DL_\alpha PQL^n) = \sum_{k=1}^{n} Z_k + \sum_{p=1}^{d} \alpha_p \quad (29)$$

A partir de então, uma nova coluna gerada pelo subproblema k é inserida no *PMR* se o seu custo reduzido θ_k for positivo, isto é, $\theta_k = Z_k - \beta_k > 0$. Assim, o *PMR* coordena as soluções dos problemas por meio de suas variáveis duais, buscando uma solução para o problema original. O *PMR* inicial é gerado por meio de uma heurística baseada na apresentada em Beasley (1998). Em seguida, novas colunas são geradas até que um critério de parada seja satisfeito, e o *PMR* final, formado por todas as colunas geradas, é resolvido de forma inteira ($\lambda_{sk} \in \{0,1\}$), e consequentemente, sua solução será equivalente a uma solução viável para o *PQL*, e indiretamente para o *PQ*. Por convenção, esse problema será tratado como PMR_{PLI}, e o valor de sua solução como $v(PMR_{PLI})$. Logo, a solução do *PMR* de forma inteira - $v(PMR_{PLI})$ - apresenta uma solução viável para o *PQ*, e a solução do *PMR* por meio de sua relaxação linear - $v(PMR_{PL})$ - e a solução da decomposição lagrangeana - $v(DL_\alpha PQL^n)$ - apresentam limitantes para o *PQ*.

3. Outros limitantes para o *PQ*

O valor da solução da relaxação linear do *PQL* (substituir a restrição 7 por $0 \le x_i \le 1$) é um limitante trivial para o *PQ*, e é conhecido como *roof dual* [Billionnet e Calmels 1996]. Por notação, esse modelo será tratado como $\overline{PQL}$. Outro limitante para o *PQ* pode ser obtido por meio da inserção de uma restrição de corte de *Chvatal-Gomory* (30) em *PQL*. Essa restrição é baseada em uma das restrições de corte apresentadas em Billionnet *et al.* (1999) para o Problema Quadrático da Mochila - *PQM*. Por convenção, o modelo *PQL* com essa restrição será tratado como *PQLC*.

$$x_i + x_j + x_k - w_{ij} - w_{jk} - w_{ik} \le 1 \; i < j < k, q_{ij} \ne 0, q_{jk} \ne 0, q_{ik} \ne 0 \quad (30)$$

A relaxação lagrangeana das restrições (3), (4) e/ou (5) também pode ser utilizada para obtenção de outros limitantes para o *PQ*. Essas restrições podem ser combinadas de forma a gerar diferentes modelos para o problema, sendo que a minimização de cada dual correspondente apresenta um limitante para o *PQ*.

4. Resultados computacionais

Os resultados apresentados a seguir foram obtidos por meio de um conjunto de 45 instâncias disponíveis na *OR-Library*. Essas instâncias são separadas em 6 classes (A, B, C, D, E e F) com diferentes características. Essas instâncias estão entre as mais difíceis encontradas na literatura [Glover *et al.* 1998].

Para resolver a relaxação linear do problema ($\overline{PQL}$) e a relaxação linear com a restrição de corte (*PQLC*), foi utilizado o CPLEX 10.0.1 [Ilog 2006]. O dual lagrangeano das relaxações lagrangeanas tradicionais citadas na seção anterior foi otimizado por meio

do algoritmo de subgradiente apresentado em Narciso e Lorena (1999). O CPLEX também foi utilizado para resolver os problemas de forma exata a cada iteração do algoritmo de subgradiente. No método proposto (*DecLagGC*), a divisão do grafo *G* foi realizada com o uso da biblioteca de heurísticas METIS [Karypis e Kumar, 1998], que particiona o grafo do problema minimizando o número de arestas com terminações em *clusters* distintos. O *PMR* e os subproblemas também foram resolvidos pelo CPLEX. Todos esses métodos foram executados com o tempo limite de 1 hora de processamento para cada instância.

Tabela 1. *Gaps* (%) obtidos pelos limitantes apresentados.

Inst.	*m*	*Dsde* (%)	$\overline{PQL}$	*PQLC*	*Lag*	*DecLagGC*	
						$DL_{\alpha}PQL^{n}$	PMR_{PL}
A	30-100	6,25-50	9,99	6,16	2,98	2,22	0,17
B	20-125	10-100	656,34	404,22	657,66	323,72	3,44
C	40-100	10-80	33,31	15,11	9,71	3,41	0,03
D	100	10-100	155,37	85,35	111,40	109,87	0,11
E	200	10-50	160,56	95,43	136,72	713,29	0,05
F	500	10-100	159,28	126,12	124,66	1102,40	0,38
Tempo médio (seg.)			**173,96**	**371,90**	**2725,84**	**1282,81**	

A Tabela 1 apresenta os *gaps* obtidos pelos métodos apresentados em relação às melhores soluções conhecidas [Glover *et al.* 1998]. A coluna *Dsde* apresenta as densidades da matriz **Q** das instâncias de cada classe, e a coluna *Lag* apresenta os melhores *gaps* obtidos pelas relaxações lagrangeanas tradicionais. Os limitantes ($DL_{\alpha}PQL^{n}$ e PMR_{PL}) obtidos pelo método proposto apresentam excelentes *gaps* para praticamente todas as instâncias. O número de *clusters* (*n*) utilizado variou de 2 a 20, de acordo com a instância.

Tabela 2. Resultados obtidos para as instâncias das classes B e F.

Inst.	*m*	*Dsde* (%)	Melhores soluções	Glover *et al.* (2002) – *Perc* (%)					*DecLagGC*		
				DDT	*A2n*	*A2t*	*V3n*	*V3t*	*n*	$v(PMR_{PLI})$	*Perc*
1b	20	100	133	73,7	**100**	78,9	10,5	10,5	2	133	**100**
2b	30	100	121	95,0	75,2	86,8	5,0	5,0	2	121	**100**
3b	40	100	118	47,5	86,4	80,5	0,0	0,0	2	118	**100**
4b	50	100	129	66,7	78,3	78,3	0,0	0,0	2	129	**100**
5b	60	100	150	70,0	**100**	60,0	0,7	0,7	2	150	**100**
6b	70	100	146	43,2	77,4	72,6	2,7	2,7	2	128	**87,7**
7b	80	100	160	56,3	**100**	**100**	0,6	0,6	2	160	**100**
8b	90	100	145	61,4	80,7	80,7	0,0	0,0	2	145	**100**
9b	100	100	137	70,1	92,7	75,9	0,0	0,0	2	133	**97,1**
10b	125	10	154	65,6	78,6	78,6	0,0	0,0	2	147	**95,5**
1f	500	10	61194	99,2	77,9	76,9	92,9	92,8	20	60959	**99,6**

A Tabela 2 apresenta as soluções inteiras obtidas para as instâncias das classes B e F. Nessa tabela, a coluna *Perc* apresenta o percentual da solução obtida em relação a melhor solução conhecida ($(v(PMR_{PLI})/v(OPT))*100$). Os resultados obtidos pela *DecLagGC* são comparados aos resultados apresentados pelas diferentes heurísticas apresentadas em Glover *et al.* (2002). Como pode ser observado, o método proposto apresentou melhores resultados para todas as instâncias da classe B (com tempo médio de 96,42 seg.) e para a instância da classe F com densidade de 10% (em 5585,00 seg.). Para as demais, a *DecLagGC* não foi capaz de encontrar uma solução em um tempo aproximado de 1 hora. O método descrito em Billionnet e Elloumi (2007) não foi capaz de resolver as instâncias da classe F, e segundo os autores, todas as soluções ótimas foram encontradas para as instâncias da classe B com um limite de tempo de 2 horas.

A *DecLagGC* obteve um *gap* médio de 0,11% para as instâncias das classes D com tempo médio de processamento foi de 4522,32 segundos, enquanto que o apresentado por Billionnet e Elloumi (2007) foi de 7,57% com tempo médio de 10 minutos por instância. Para as instâncias das classes A e C, o método proposto por Billionnet e Elloumi (2007) obteve as soluções ótimas em um tempo limite de 2 horas por instância, porém não foi capaz de resolver os problemas da classe E. Já a *DecLagGC* obteve as soluções ótimas para todas as instâncias das classes A, C e E, com o tempo médio de 96,40 e 64,13 segundos para as instâncias das classes A e C, e da classe E, respectivamente. Glover *et al.* (2002) não apresentam resultados para essas instâncias.

5. Conclusões

Este trabalho apresentou uma nova estratégia de decomposição lagrangeana baseada na técnica de geração de colunas para resolver o problema de programação quadrática binária irrestrita. O método proposto, além de encontrar soluções viáveis, também apresenta duas alternativas para obtenção de limitantes para o *PQ*. Instâncias de difícil solução e com diferentes características foram utilizadas para avaliar o método proposto.

Os resultados apresentam indícios de que a decomposição proposta é superior à proposta por Chardaire e Sutter (1995), pois soluções de boa qualidade foram obtidas para problemas maiores (com densidades variadas). Além disso, o método proposto foi comparado diretamente com outros métodos propostos recentemente, e apresentou excelentes resultados para praticamente todas as instâncias.

References

Beasley, J.E. (1998). "Heuristic algorithms for the unconstrained binary quadratic programming problem". *Technical Report - Management School, Imperial College*, London, UK.

Billionnet, A.; Calmels, F. (1996). "Linear programming for the 0-1 quadratic knapsack problem". *European Journal of Operational Research*, 92(2): 310-325.

Billionnet, A.; Elloumi, S. (2007). "Using a mixed integer quadratic programming solver for the unconstrained quadratic 0-1 problem". *Mathematical Programming*, 109: 55-68.

Billionnet, A.; Faye, A.; Soutif, E. (1999). "A new upper bound for the 0-1 quadratic knapsack problem". *European Journal of Operational Research*, 112: 664-672.

Chardaire, P.; Sutter, A. (1995). "A decomposition method for quadratic zero-one programming". *Management Science*, 41(4): 704-712.

Glover, F.; Alidaee, B.; Rego, C.; Kochenberger, G. (2002). "One-pass heuristics for large-scale unconstrained binary quadratic problems". *European Journal of Operational Research*, 137(2): 272-287.

Glover, F.; Kochenberger, G.A.; Alidaee, B. (1998). "Adaptative memory tabu search for binary quadratic programs". *Management Science*, 44(3): 336-345.

Glover, F.; Woolsey, E. (1974). "Converting a 0-1 polynomial programming problem to a 0-1 linear program". *Operations Research*, 22: 180-182.

Ilog (2006). "ILOG CPLEX 10.0: user's manual". France, 478 p.

Karypis, G.; Kumar, V. (1998). "Multilevel k-way partitioning scheme for irregular graphs". *Journal of Parallel and Distributed Computing*, 48: 96-129.

Narciso, M.G.; Lorena, L.A.N. (1999). "Lagrangean/surrogate relaxation for generalized assignment problems". *European Journal of Operational Research*, 114: 165-177.