UM ALGORITMO EXATO PARA O PROBLEMA DA DIVERSIDADE MÁXIMA

João Leonardo Ribeiro Neto
PUC Minas - Instituto de Ciências Exatas e Informática
Av. Dom José Gaspar, 500 - Coração Eucarístico
CEP: 30535-901 - Belo Horizonte - Minas Gerais - Brasil
joaoneto@pucminas.br

Geraldo Paulino M. Pereira
PUC Minas - Instituto de Ciências Exatas e Informática
Av. Dom José Gaspar, 500 - Coração Eucarístico
CEP: 30535-901 - Belo Horizonte - Minas Gerais - Brasil
paulino@pucminas.br

Bruno Takane
UFMG - Programa de Pós Graduação em Engenharia de Produção
Av. Antônio Carlos, 6627 - Pampulha
CEP: 31270-901 - Belo Horizonte - Minas Gerais - Brasil
brtakane@gmail.com

Gilberto Miranda Junior
UFMG - Programa de Pós Graduação em Engenharia de Produção
Av. Antônio Carlos, 6627 - Pampulha
CEP: 31270-901 - Belo Horizonte - Minas Gerais - Brasil
miranda@dep.ufmg.br

Ricardo Saraiva de Camargo
UFMG - Programa de Pós Graduação em Engenharia de Produção
Av. Antônio Carlos, 6627 - Pampulha
CEP: 31270-901 - Belo Horizonte - Minas Gerais - Brasil
rcamargo@dep.ufmg.br

Resumo

O termo diversidade está relacionado à variedade de características, idéias ou elementos diferentes entre si dentro de um determinado contexto, sendo importante para o pluralismo, heterogeneidade, tolerância mútua e sobrevivência de idéias. Na área de otimização combinatória, o Problema da Diversidade Máxima (PDM) consiste em selecionar um subconjunto de m elementos de um conjunto de n elementos, de tal forma que a diversidade entre os seus elementos selecionados seja máxima. Neste trabalho é apresentado uma nova formulação para este problema baseado na Técnica de Reformulação/Linearização. Devido às características da formulação proposta e da dificuldade de resolução, o método de Decomposição de Benders Revisado é aplicado ao problema, assim como uma técnica de pré-processamento de modo a acelerar a sua convergência. Testes são realizados para avaliar o desempenho do método aplicado ao problema e em seguida, uma análise é feita comparando-o com outro algoritmo descrito na literatura. Os resultados computacionais mostram que o método proposto demonstra ser competitivo frente aos métodos exatos descritos na literatura.

Palavras chave: Problema da Diversidade Máxima; Método de Decomposição de Benders; Técnica de Reformulação/Linearização.

Abstract

The term diversity is related to the variety of features, ideas, or different elements together within a given context, it is important to pluralism, diversity, tolerance and mutual survival of ideas. In the area of combinatorial optimization,the Maximum Diversity Problem (MDP) is to select a subset m elements of a set of n elements, such that the diversity among the elements selected is maximum. This paper presents a new formulation for this problem based on the Reformulation-Linearization Technique. Due to the characteristics of the proposed formulation and the difficulty of resolution, the method Revised Benders Decomposition is applied to the problem, as well as a preprocessing technique to accelerate the convergence. Tests are performed to evaluate the performance of the method applied to the problem and then analysis is made by comparing it with another algorithm described in the literature. The computation results show that the proposed method proves to be competitive compared to the exact methods described in literature.

Keywords: Maximum Diversity Problem; Benders Decomposition method; Reformulation-Linearization Technique.

1 Introdução

Na área de otimização combinatória, o Problema da Diversidade Máxima (PDM) consiste em selecionar um subconjunto de m elementos de um conjunto de n elementos, de tal forma que a diversidade entre os seus elementos selecionados seja máxima. Neste caso, o termo diversidade é usualmente substituído por distância entre elementos, em que esta distância é adequada para cada aplicação específica. Em função do potencial de aplicações do PDM e dos seus de benefícios econômicos, seria interessante ter-se a melhor seleção (solução ótima) de elementos para que possa justificar o investimento realizado. Desse modo, o principal objetivo deste trabalho é apresentar uma formulação adequada e um método exato eficiente com geração de planos de cortes para a solução do PDM. De modo a comprovar a eficiência do algoritmo, testes serão realizados a partir de instâncias conhecidas na literatura.

sectionDefinição do problema da diversidade máxima O PDM consiste em selecionar um subconjunto M com m elementos de um conjunto N, onde N = {1,. . . ,n}, de tal forma que a diversidade entre os seus elementos selecionados seja máxima. Para que o PDM tenha significado, deve-se assumir que 2 $\leq$ m $\leq$ n. No PDM, é comum representar a diversidade como uma distância d_{ij} entre dois elementos i, j $\in$ N. Esta distância é calculada como uma métrica normalizada entre os atributos dos elementos de N. Então, sejam R o conjunto de atributos, onde R = {1,. . . ,r} e a_{ik} o estado ou valor do atributo k $\in$ R para o elemento i $\in$ N, sendo que a_{ik} pode real ou inteiro. Uma distância muito usada (Ghosh, J., 1996) é dado pela norma-p:

$$d_{ij} = \sqrt[p]{\sum_{k \in R} (|a_{ik} - a_{jk}|)^p} \tag{1}$$

Em que a quantidade d_{ij} representa a distância, ou dessemelhança, entre os elementos i, j $\in$ N usando uma determinada métrica, onde $d_{ij} > 0$ para i $\neq$ j e d_{ij} = 0, caso contrário. Deste modo, o valor da diversidade total ζ(M) para um dado subconjunto M, pode ser medido como a soma das distâncias entre cada par distinto de elementos é dado por:

$$\zeta(M) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left[\sqrt[p]{\sum_{k \in R} (|a_{ik} - a_{jk}|)^p} \right] \tag{2}$$

Como discutido em Kuo, C. et al. (1993), uma métrica pode ser mais adequada que a outra, dependendo da aplicação. Se por exemplo, for considerada a distância euclidiana para o cálculo da diversidade d_{ij} dada pela equação 1, temos, então:

$$d_{ij} = \sqrt{\sum_{k \in R} (a_{ik} - a_{jk})^2} \tag{3}$$

E se for considerada distância euclidiana para o cálculo da diversidade total ζ(M) dada pela equação 2 para um dado conjunto M, temos:

$$\zeta(M) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left[\sqrt{\sum_{k \in R} (a_{ik} - a_{jk})^2} \right] \tag{4}$$

1.1 Trabalhos relacionados

Um das primeira abordagens a caracterizar o PDM como um problema de otimização em um contexto de recursos genéticos encontra-se em Glover, F. et al. (1977). Em Kuo, C. et al. (1993) discutem-se vários contextos em que o problema pode ser levantado, como planejamento de mercado, seleção de portifólio e formação de comitê. Eles mostraram que o problema é NP-difícil e apresentaram duas definições diferentes para o problema utilizando formulações de programação linear inteira mista, os modelos max-sum e max-min. A primeira tem como objetivo maximizar a soma das diversidades de um subconjunto de elementos selecionados a partir de um conjunto maior. Enquanto que a última procura selecionar um subconjunto de elementos de um conjunto maior, de modo que a distância mínima entre os elementos selecionados seja maximizada. Neste trabalho será abordada a definição de max-sum e por isso, será feita uma revisão de literatura seguindo apenas essa definição. Em Aringhieri, R. et al. (2008) os autores citam, ainda, a aplicação durante a formação de equipes de trabalho, júris e grupos de estudantes para trabalho em projetos, pois é comum desejar um número fixo de indivíduos cujas características sejam a mais diversificada quanto possível (Fernandez, J., 1991), (Cox, T., 1993). Muitos trabalhos na literatura tratam o PDM através de heurísticas e métodos exatos, como Kuo, C. et al. (1993); Glover, F. et al. (1998); Ghosh, J. (1996); Andrade, P. et al. (2003); Silva, G. et al. (2004, 2007); Duarte, A. & Marti, R. (2007); Palubeckis, G. (2007); Erkut, E. & Neuman, S. (1991) dentre outros.

2 Formulação Matemática e Algoritmo proposto

Nesta Seção, uma nova formulação baseada na Técnica de Reformulação/Linearização e um algoritmo baseado no método de Decomposição de Benders Revisado são propostos para solução do PDM.

2.1 Aplicação da técnica TRL

Para apresentar uma nova formulação para o PDM, considera-se a formulação original (Glover, F. et al., 1998) do PDM proposta por Kuo, C. et al. (1993) com região viável $X = \{x_i, i \in N : \sum_{i \in N} x_i = m, x_i \in \{0,1\}, \forall i \in N\}$:
Formulação F1

$$\max \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} x_i x_j \tag{5}$$

$$\text{s.a:} \sum_{i \in N} x_i = m \tag{6}$$

$$x_i \in \{0,1\}, \qquad \forall\, i \in N \tag{7}$$

A formulação (5) – (7) é não-linear o que torna sua relaxação linear (RL) fraca. É possível obter uma formulação com uma RL melhor, mas com mais variáveis e restrições, usando a técnica TRL.Para isso, aplica-se o método TRL - nível $d = 1$ sobre a região viável $X_0 = X$, onde cada igualdade é multiplicada pelas variáveis binárias $x_j, \forall j \in N$, e cada desigualdade, incluindo as restrições de domínio das variáveis, são multiplicadas tanto pelas variáveis binárias x_j quanto por seus complementos $(1 - x_j), \forall j \in N$. Dessa forma, variáveis auxiliares lineares y_{ij} são criadas ao substituí-las por cada termo não-linear $x_i x_j, \forall i, j \in N$. Após a aplicação da TRL sobre a formulação **F1** equações (5) – (7), obtem-se a formulação **FB1** com o mesmo número de variáveis binárias que **F1**, mas com uma quantidade maior de variáveis lineares.

Formulação FB1

$$\max \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} y_{ij} \tag{8}$$

s.a: restrições (6) − (7)

$$\sum_{j:i<j} y_{ij} + \sum_{j:i>j} y_{ji} = (m-1)x_i \qquad \forall\, i \in N \tag{9}$$

$$y_{ij} \geq x_i + x_j - 1 \qquad \forall\, i,j \in N : i < j \tag{10}$$

$$y_{ij} \leq x_i \qquad \forall\, i,j \in N : i < j \tag{11}$$

$$y_{ij} \leq x_j \qquad \forall\, i,j \in N : i < j \tag{12}$$

$$y_{ij} \geq 0, \qquad \forall\, i,j \in N : i < j \tag{13}$$

A formulação acima é claramente um problema linear inteiro misto. Ela tem uma RL superior conforme Adams,W. P. & Sherali, H. D. (1990), mas também uma quantidade maior de restrições e variáveis lineares. Ela pode ser fortalecida ainda mais através da aplicação do método TRL - nível $d = 2$ sobre a região viável $X_1 = \{x_i, y_{ij}, i, j \in N, i < j$: restrições$(6) - (6), (9) - (13)\}$, onde cada igualdade é multiplicada pelas variáveis binárias $x_k, \forall k \in N$, e cada desigualdade, incluindo as restrições de domínio das variáveis, são multiplicadas tanto pelas variáveis binárias x_k quanto por seus complementos $(1 - x_k), \forall k \in N$. Dessa forma, variáveis auxiliares lineares z_{ijk} são criadas ao substituí-las por cada termo não-linear $y_{ij}x_k, \forall i, j, k \in N$. Assim, nove famílias de desigualdades válidas são adicionadas à formulação construindo **FB2**:

Formulação FB2

$$\max \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} y_{ij} \tag{14}$$

s.a: restrições (6) − (7), (9) − (13)

$$\sum_{i:i<j<k} z_{ijk} + \sum_{i:j<i<k} z_{jik} + \sum_{i:j<k<i} z_{jki} = (m-2)y_{jk} \qquad \forall\, j,k \in N : j < k \tag{15}$$

$$z_{ijk} \leq y_{ij} \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{16}$$

$$z_{ijk} \leq y_{ik} \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{17}$$

$$z_{ijk} \leq y_{jk} \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{18}$$

$$-z_{ijk} + y_{ik} + y_{jk} - x_k \leq 0 \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{19}$$

$$-z_{ijk} + y_{jk} + y_{ij} - x_j \leq 0 \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{20}$$

$$-z_{ijk} + y_{ik} + y_{ij} - x_i \leq 0 \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{21}$$

$$x_i + x_j + x_k - y_{ij} - y_{jk} - y_{ik} + z_{ijk} \leq 1 \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{22}$$

$$z_{ijk} \geq 0 \qquad \forall\, i,j,k \in N : i < j,\ j < k \tag{23}$$

Pode-se observar que **FB2** tem o mesmo número de variáveis binárias que **F1** e **F2** e uma quantidade maior ainda de variáveis lineares que a formulação **FB1**, ou seja, uma quantidade muito maior de variáveis lineares que binárias, o que torna interessante usar o método de Decomposição de Benders Revisado.

2.2 Aplicação do método de Decomposição de Benders Revisado

Para a aplicação do método de Decomposição de Benders Revisado ao PDM, observa-se que todas as familias de desigualdades geradas, com exceção de (15), têm uma estrutura especial, ou seja, estão escritas na forma $\forall i, j, k \in N, i < j, j < k$. Apesar da restrição (15) contribuir muito para a formulação, ela a torna mais difícil de resolver e portanto não vamos considerá-la para solução do problema da região de viabilidade $Z = \{z_{ijk}, \forall i, j, k \in N, i < j, j < k : \text{s.a } (16) - (22)\}$, optando-se por usar uma representação parcial do modelo gerado através do TRL - nível d = 2, conforme já sugerido por Adams,W. P. & Sherali, H. D. (1986). Para qualquer $\bar{x}_i, \bar{y}_{ij} \in X_1, \forall i, j \in N, i < j$, onde $X_1 = \{x_i, y_{ij}, \forall i, j \in N, i < j : \text{s.a: } (6) - (7), (9) - (13)\}$, uma das duas afirmações seguintes é verdadeira:

1. $\exists z_{ijk} \in Z, \forall i, j, k \in N, i < j < k, \text{onde} Z = \{(16) - (23)\}$ ou
2. $\nexists z_{ijk} \in Z, \forall i, j, k \in N, i < j < k.$

Para o caso 1 implica que existe um $z_{ijk} \in Z, \forall i, j, k \in N, i < j, j < k$, onde Z é a região de viabilidade de z_{ijk}. Já o caso 2 implica que não existe um $z_{ijk} \in Z, \forall i, j, k \in N, i < j, j < k$ e pelo *Lema de Farkas* pode-se achar um hiperplano que separa o ponto $\bar{x}_i, \bar{y}_{ij} \in X_1, \forall i, j \in N, i < J$ da região Z. Uma forma de achá-lo é através da introdução de variáveis de erro e que viabilizariam o subsistema:

$$\min e_1 + e_2 + e_3 + e_4 + e_5 + e_6 + e_7 \tag{24}$$
$$\text{s.a: } -z_{ijk} + e_1 \geq -\bar{y}_{ij} \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{25}$$
$$-z_{ijk} + e_2 \geq -\bar{y}_{ik} \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{26}$$
$$-z_{ijk} + e_3 \geq -\bar{y}_{jk} \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{27}$$
$$z_{ijk} + e_4 \geq \bar{y}_{ik} + \bar{y}_{jk} - \bar{x}_k \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{28}$$
$$z_{ijk} + e_5 \geq \bar{y}_{jk} + \bar{y}_{ij} - \bar{x}_j \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{29}$$
$$z_{ijk} + e_6 \geq \bar{y}_{ik} + \bar{y}_{ij} - \bar{x}_i \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{30}$$
$$-z_{ijk} + e_6 \geq \bar{x}_i + \bar{x}_j + \bar{x}_k - \bar{y}_{ij} - \bar{y}_{jk} - \bar{y}_{ik} - 1 \qquad \forall i, j, k \in N : i < j, j < k \tag{31}$$
$$e_1, e_2, e_3, e_4, e_5, e_6, e_7 \geq 0 \tag{32}$$
$$z_{ijk} \geq 0 \qquad \forall\, i, j, k \in N : i < j,\ j < k \tag{33}$$

Associando variáveis duais u_1 - u_7 às restrições (25) - (31), pode-se encontrar o hiperplano através do subproblema dual escrito como:

$$\begin{aligned}
\max\ & -\bar{y}_{ij}u_1 - \bar{y}_{ik}u_2 - \bar{y}_{jk}u_3 \\
& + (\bar{y}_{ik} + \bar{y}_{jk} - \bar{x}_k)u_4 + (\bar{y}_{jk} + \bar{y}_{ij} - \bar{x}_j)u_5 + (\bar{y}_{ik} + \bar{y}_{ij} - \bar{x}_i)u_6 \\
& + (\bar{x}_i + \bar{x}_j + \bar{x}_k - \bar{y}_{ij} - \bar{y}_{jk} - \bar{y}_{ik})u_7 && (34) \\
\text{s.a:}\ & -u_1 - u_2 - u_3 + u_4 + u_5 + u_6 - u_7 \leq 0 && (35) \\
& u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 \leq 1 && (36) \\
& u_1, u_2, u_3, u_4, u_5, u_6, u_7 \geq 0 && (37)
\end{aligned}$$

Para qualquer $\bar{x}_i, \bar{y}_{ij} \in X_1, \forall i, j \in N, i < j$, tal que não existe $z_{ijk} \in Z, \forall i, j, k \in N, i < j, j < k$, pode-se achar um raio extremo que provê um novo corte para o problema mestre **PM**, tal que (34) > 0. Desse modo, fixada uma solução $(x, y) = (\bar{x}, \bar{y})$, bastaria resolver o dual em $u_1 - u_7$ para obter um corte violado. No entanto, observando a estrutura das restrições (16) – (22), pode-se notar que a variável z_{ijk} está canalizada, ou seja, pode-se reescrever os pares de desigualdades (16) e (19), (17) e (20), (18) e (21) e (22) e (23), na forma das desigualdades (38), (39), (40) e (41), respectivamente:

$$\begin{aligned}
\bar{y}_{ik} + \bar{y}_{jk} - \bar{x}_k \leq z_{ijk} \leq \bar{y}_{ij} && (38) \\
\bar{y}_{jk} + \bar{y}_{ij} - \bar{x}_j \leq z_{ijk} \leq \bar{y}_{ik} && (39) \\
\bar{y}_{ik} + \bar{y}_{ij} - \bar{x}_i \leq z_{ijk} \leq \bar{y}_{jk} && (40) \\
0 \leq z_{ijk} \leq \bar{y}_{ij} + \bar{y}_{jk} + \bar{y}_{ik} - \bar{x}_i - \bar{x}_j - \bar{x}_k + 1 && (41)
\end{aligned}$$

Algorithm 1 Cálculo do valor de cada variável dual

$\forall i, j, k \in N : i < j, j < k$
Se $(\bar{y}_{ik} + \bar{y}_{jk} - \bar{x}_k > \bar{y}_{ij})$, então $u_1 = u_4 = 1/7$. Senão $u_1 = u_4 = 0$.
Se $(\bar{y}_{jk} + \bar{y}_{ij} - \bar{x}_j > \bar{y}_{ik})$, então $u_2 = u_5 = 1/7$. Senão $u_2 = u_5 = 0$.
Se $(\bar{y}_{ik} + \bar{y}_{ij} - \bar{x}_i > \bar{y}_{jk})$, então $u_3 = u_6 = 1/7$. Senão $u_3 = u_6 = 0$.
Se $(\bar{y}_{ij} + \bar{y}_{jk} + \bar{y}_{ik} - \bar{x}_i - \bar{x}_j - \bar{x}_k + 1 < 0)$, então $u_7 = 1/7$. Senão $u_7 = 0$.

Ou seja, se por exemplo, $(\bar{y}_{ik} + \bar{y}_{jk} - \bar{x}_k > \bar{y}_{ij})$, então o par de desigualdades (16) e (19) seriam desrespeitados, o que implicaria em penalidade nas variáveis u_1 e u_4 no subproblema dual (34) - (37). Se caso todas os pares de desigualdades (16) e (19), (17) e (20), (18) e (21) e (22) e (23) fossem desrespeitados, então, conforme a restrição (36), a soma das variáveis duais deve ser menor ou igual a 1 e, portanto, o valor máximo de cada variável dual $u_1 - u_7$ seria igual a $1/7$, pois desta forma o subsistema (35) - (37) continuaria sendo viável e um novo corte $-\bar{u}_{1ijk}y_{ij} - \bar{u}_{2ijk}y_{ik} - \bar{u}_{3ijk}y_{jk} + \bar{u}_{4ijk}(y_{ik} + y_{jk} - x_k) + \bar{u}_{5ijk}(y_{jk} + y_{ij} - x_j) + \bar{u}_{6ijk}(y_{ik} + y_{ij} - x_i) + \bar{u}_{7ijk}(x_i + x_j + x_k - y_{ij} - y_{jk} - y_{ik}) \leq 0$ poderia ser adicionado ao

Problema Mestre Relaxado (**PMR**), onde as desigualdades são adicionadas sob demanda até um número H máxmimo de desigualdades eficientemente utilizáveis, podendo ser geradas ao longo da árvore de *Branch-and-Bound*.

2.3 Pré-processamento do método de Decomposição de Benders Revisado

De modo a acelerar a convergência do método proposto, é utilizado uma técnica de pré-processamento no método de Decomposição de Benders Revisado (McDaniel, D. & Devine, M., 1977). Ela consiste em gerar cortes a partir de soluções fracionárias, ao desconsiderar as restrições de integralidade e resolvendo o Problema Mestre durante algumas iterações. A idéia geral é de gerar cortes a partir de um programa de programação linear ao invés de programas inteiros. Existem algumas regras para determinar o número de vezes que o problema mestre deve ser resolvido com as restrições de integralidade relaxadas. Algumas possibilidades são:

- Até que não se consiga mais realizar iterações, ou seja, até quando não se perceba que o limite inferior possa ser melhorado;
- resolver o problema relaxado para as primeiras k iterações, k inteiro positivo, e então resolver o problema inteiro;
- ou então até que seja atingido um critério de convergência *delta*. Por exemplo um *GAP* de otimalidade $<$ *delta*.

3 Resultados computacionais

Todos os testes foram feitos usando um computador *Intel(R) Core(TM)* 2 *Quad CPU Q*8400 2.66*Ghz* com 4*GB* de memória *RAM* e sistema operacional *Linux*. Foram utilizados dois conjuntos de instâncias para os testes:

- **GKD**: propostos por Marti, R. et al. (2010), consistem em 50 matrizes cujos valores são distâncias euclidianas entre cada par de pontos com coordenadas geradas aleatoriamente no intervalo $[0, 10]$, onde cada ponto tem r coordenadas variando entre 2 e 21. Os tamanhos das matrizes são tais que $n = 25$ para $m = 2$ e $m = 7, n = 50$ para $m = 5$ e $m = 15, n = 100$ para $m = 10$ e $m = 30, n = 125$ para $m = 12$ e $m = 37$ e $n = 150$ para $m = 15$ e $m = 45$.
- **SOM**: propostos por Silva, G. et al. (2004), consistem em 50 matrizes cujos valores inteiros são gerados seguindo uma distribuição uniforme no intervalo $[0, 9]$. Os tamanhos das matrizes são os mesmos para o conjunto de instâncias **GKD**.

O *GAP* de otimalidade é calculado como o limite superior (retornada pelo *CPLEX*) menos o limite inferior, dividido pelo limite superior e multiplicado por 100, ambos retornados pelo *CPLEX*). Para o conjunto **GKD** não foi estabelecido um tempo limite como critério de parada para o algoritmo, enquanto que para o conjunto **SOM** estabeleceu-se um limite de execução igual a uma hora (3600*s*), devido às suas instâncias serem caracterizadas por uma baixa variabilidade e portanto com um grande número de soluções degeneradas, o que desfavorece a utilização de um método exato para a sua solução, enquanto beneficia heurísticas a encontrarem bons limites superiores.

As tabelas 1 e 2 mostram o número de iterações de pré-processamento **NHS** (seção 2.3), aplicados a cada instância, o tempo de execução em segundos **T(s)** e o **GAP** de otimalidade, durante a fase de pré-processamento. As instâncias que não tiveram iteração de pré-processamento estão marcadas com o simbolo "-". Para o conjunto **GKD**, os resultados mostram que as 25 instâncias que tiveram iterações de pré-processamento, em um total de 50, obtiveram **GAP** de otimalidade inferior a 2% com exceção da instância 17, sendo que 14(28%) instâncias atingiram a otimalidade apenas nesta fase de pré-processamento. Apenas 5 instâncias gastaram mais uma hora de tempo computacional nesta fase.

Tabela 1: Resultados de pré-processamento para as instâncias **GKD** propostas por Marti, R. et al. (2010) sem tempo limite como critério de parada

Instância	n	m	NHS	T(s)	GAP(%)	Instância	n	m	NHS	T(s)	GAP(%)
1	25	2	0	-	-	26	100	30	3	285,13	0,25
2	25	2	0	-	-	27	100	30	3	820,76	0,02
3	25	2	0	-	-	28	100	30	2	54,59	0,00
4	25	2	0	-	-	29	100	30	3	147,31	0,00
5	25	2	0	-	-	30	100	30	3	390,49	0,00
			Média	-	-				Média	339,66	0,05
6	25	7	2	0,05	0,00	31	125	12	0	-	-
7	25	7	2	0,03	0,00	32	125	12	0	-	-
8	25	7	2	0,07	0,00	33	125	12	0	-	-
9	25	7	2	0,03	0,00	34	125	12	0	-	-
10	25	7	2	0,04	0,00	35	125	12	0	-	-
			Média	0,044	0,00				Média	-	-
11	50	5	0	-	-	36	125	37	3	3003,99	0,78
12	50	5	0	-	-	37	125	37	3	1419,36	0,00
13	50	5	0	-	-	38	125	37	3	3219,65	0,17
14	50	5	0	-	-	39	125	37	4	15562,09	0,07
15	50	5	0	-	-	40	125	37	4	1837,3	0,27
			Média	-	-				Média	5008,48	0,26
16	50	15	3	4,13	0,78	41	150	15	0	-	-
17	50	15	2	1,32	10,23	42	150	15	0	-	-
18	50	15	3	3,05	0,00	43	150	15	0	-	-
19	50	15	3	3,52	0,00	44	150	15	0	-	-
20	50	15	5	6,45	0,93	45	150	15	0	-	-
			Média	3,69	2,39				Média	-	-
21	100	10	0	-	-	46	150	45	4	35514,69	0,00
22	100	10	0	-	-	47	150	45	3	15066,3	0,00
23	100	10	0	-	-	48	150	45	2	8259,55	1,37
24	100	10	0	-	-	49	150	45	3	11024,29	0,00
25	100	10	0	-	-	50	150	45	2	3811,85	1,48
			Média	-	-				Média	14735,34	0,57

Para o conjunto **SOM**, entre as 30 instâncias que tiveram iterações de pré-processamento, em um total de 50, apenas 4 atingiram a otimalidade nesta fase de pré-processamento. Para as instâncias restantes, obteve-se **GAP** de otimalidade variando entre $4,57\%$ e $13,64\%$ para tamanhos entre $n = 25$ para $m = 7$ e $n = 50$ para $m = 15$. Para os de tamanho entre $n = 100$ para $m = 30$ e $n = 150$ para $m = 45$, o **GAP** de otimalidade variou entre $13,43\%$ e $24,43\%$. Observa-se para este conjunto que 11 instâncias gastaram uma hora de tempo computacional nesta fase.

As tabelas 3 e 4 fazem comparação entre o algoritmo *Branch-and-Bound* (**BB**) proposto por Marti, R. et al. (2010) e **DBR** com um limite de execução igual a uma hora, uma vez que eles reportam os seus resultados considerando esse limite. Além disso, eles mostram apenas a média do tempo computacional em segundos e do **GAP** de otimalidade para cada combinação de valores de n e m. Para **GKD**, observa-se que o algoritmo **DBR** atinge a

Tabela 2: Resultados de pré-processamento para as instâncias **SOM** propostas por Silva, G. et al. (2004), considerando um tempo limite de execução igual a uma hora (3600*s*) como critério de parada

Instância	n	m	NHS	T(s)	GAP(%)	Instância	n	m	NHS	T(s)	GAP(%)
1	25	2	0	-	-	26	100	30	4,00	1235,63	14,84
2	25	2	0	-	-	27	100	30	4,00	1338,66	15,16
3	25	2	0	-	-	28	100	30	4,00	1841,97	13,43
4	25	2	0	-	-	29	100	30	4,00	1392,33	14,87
5	25	2	0	-	-	30	100	30	4,00	3600,00	15,81
			Média	-	-				Média		14,82
6	25	7	2	0,09	0,00	31	125	12	0,00	-	-
7	25	7	3	0,15	0,00	32	125	12	0,00	-	-
8	25	7	3	0,12	0,00	33	125	12	0,00	-	-
9	25	7	2	0,09	4,57	34	125	12	0,00	-	-
10	25	7	3	0,11	0,00	35	125	12	0,00	-	-
			Média	0,11	0,91				Média	-	-
11	50	5	0	-	-	36	125	37	3,00	3600,00	17,96
12	50	5	0	-	-	37	125	37	3,00	3600,00	19,04
13	50	5	0	-	-	38	125	37	3,00	3600,00	18,14
14	50	5	0	-	-	39	125	37	3,00	3600,00	18,92
15	50	5	0	-	-	40	125	37	3,00	3600,00	17,98
			Média	-	-				Média		18,41
16	50	15	2	2,04	13,47	41	150	15	4,00	468,90	24,43
17	50	15	2	2,88	9,27	42	150	15	4,00	601,30	23,35
18	50	15	3	10,94	8,25	43	150	15	4,00	460,76	22,25
19	50	15	2	2,49	13,64	44	150	15	4,00	512,51	21,75
20	50	15	2	3,19	9,31	45	150	15	4,00	721,00	23,35
			Média	4,31	10,79				Média		23,03
21	100	10	0	-	-	46	150	45	2,00	3600,00	26,30
22	100	10	0	-	-	47	150	45	2,00	3600,00	25,78
23	100	10	0	-	-	48	150	45	2,00	3600,00	26,84
24	100	10	0	-	-	49	150	45	2,00	3600,00	27,60
25	100	10	0	-	-	50	150	45	2,00	3600,00	26,18
			Média	-	-				Média		26,54

otimalidade em 38 instâncias em um total de 50 com apenas uma hora de tempo computacional. Além disso, **DBR** obtém um **GAP** médio consideravelmente menor que **BB** para todos os tamanhos, sendo que o maior **GAP** médio de **DBR** é de $1,02\%$ para as instâncias de tamanho $n = 150$ e $m = 45$, contra $13,70\%$ para as instâncias de tamanho $n = 125$ e $m = 37$ de **BB**. O algoritmo **DBR** perde apenas em tempo computacional para pequenas instâncias variando de $n = 25$ e $m = 2$ a $n = 100$ e $m = 10$, mas atingindo o mesmo **GAP** de otimalidade médio em relação a **BB** para esses tamanhos. Os resultados mostram que para o conjunto **SOM**, **DBR** gasta mais tempo na média que **BB** para atingir a otimalidade para pequenas instâncias de tamanho variando entre $n = 25$ para $m = 7$ e $n = 100$ para $m = 10$. Mas para instâncias maiores, **DBR** consegue atingir um **GAP** médio relativamente menor que **BB** para um intervalo de 3600 segundos, com exceção das instâncias de tamanho $n = 125$ para $m = 12$, uma vez que **BB** consegue atingir a otimalidade com $2678,9$ segundos.

A tabela 5 mostra o tempo computacional gasto em segundos para **DBR** atingir a otimalidade para as instâncias que não conseguiram atingir o ótimo para um limite máximo de uma hora. Observa-se que **DBR** consegue atingir a otimalidade com menos de 86400 segundos (24 horas de máquina), com exceção da instância 36 que gasta quase 86000 segundos para fechar um **GAP** de $0,78\%$ e da instância 48 que gasta aproximadamente 91000 segundos para fechar um **GAP** de $1,49\%$.

Tabela 3: Comparativo entre o algoritmo *Branch-and-Bound* **BB** (Marti, R. et al., 2010) e o algoritmo **DBR** proposto para as instâncias **GKD** por Marti, R. et al. (2010), considerando um tempo limite de execução igual a uma hora (3600s) como critério de parada

			BB		DBR					BB		DBR	
Instância	**n**	**m**	**T(s)**	**GAP(%)**	**T(s)**	**GAP(%)**	**Instância**	**n**	**m**	**T(s)**	**GAP(%)**	**T(s)**	**GAP(%)**
1	25	2			0,16	0,00	26	100	30			582,42	0,00
2	25	2			0,12	0,00	27	100	30			3600	0,80
3	25	2			0,11	0,00	28	100	30			89,4	0,00
4	25	2			0,12	0,00	29	100	30			182,48	0,00
5	25	2			0,11	0,00	30	100	30			425,43	0,00
		Média	0	0,00	0,12	0,00			Média	3576,2	8,6	975,95	0,16
6	25	7			0,41	0,00	31	125	12			485,38	0,00
7	25	7			0,38	0,00	32	125	12			669,41	0,00
8	25	7			0,41	0,00	33	125	12			69,27	0,00
9	25	7			0,39	0,00	34	125	12			28,11	0,00
10	25	7			0,41	0,00	35	125	12			178,92	0,00
		Média	0	0,00	0,40	0,00			Média	297,9	0	286,22	0,00
11	50	5			1,30	0,00	36	125	37			3600	0,78
12	50	5			1,35	0,00	37	125	37			1518,5	0,00
13	50	5			1,34	0,00	38	125	37			3600	0,17
14	50	5			1,36	0,00	39	125	37			3600	0,23
15	50	5			2,43	0,00	40	125	37			3600	0,00
		Média	0	0,00	1,56	0,00			Média	3600	13,70	3183,70	0,24
16	50	15			28,01	0,00	41	150	15			381,07	0,00
17	50	15			403,55	0,00	42	150	15			1383,99	0,00
18	50	15			6,55	0,00	43	150	15			3600	1,33
19	50	15			7,21	0,00	44	150	15			461,54	0,00
20	50	15			29,12	0,00	45	150	15			1115,76	0,00
		Média	0,4	0,00	94,89	0,00			Média	1834,4	5,40	1388,47	0,27
21	100	10			17,51	0,00	46	150	45			3600	1,23
22	100	10			14,66	0,00	47	150	45			3600	0,47
23	100	10			57,01	0,00	48	150	45			3600	1,49
24	100	10			23,55	0,00	49	150	45			3600	0,43
25	100	10			14,90	0,00	50	150	45			3600	1,48
		Média	4,4	0,00	25,53	0,00			Média	3600	10,90	3600	1,02

4 Conclusão

Neste trabalho, foi proposta uma nova formulação para o Problema de Diversidade Máxima baseada na Técnica de Reformulação/Linearização. Para resolução deste problema foi apresentado um método de Decomposição de Benders Revisado com geração de cortes ao longo da árvore Branch-and-Bound. Este método contou com um pré-processamento do método que foi capaz de acelerar a sua convergência, chegando a resolver instância de tamanho n = 125 e m = 37 com menos de uma hora de processamento, além de obter um GAP de otimalidade inferior a 2% em 48% das instâncias de um conjunto, desde que o número máximo de iterações nesta estapa fosse calibrado. Além disso, contou com uma heurística capaz de prover bons limites inferiores para o método proposto em até 140 segundo, acelerando-o ainda mais. Utilizou-se também uma técnica de fixação de variáveis que foi capaz de reduzir o tamanho da formulação em até 92% das variáveis inteiras. Os resultados apresentados mostram que esse algoritmo é capaz de resolver problemas de até 150 nós com m = 15 com menos de 1 hora de processamento, demonstrando ser competitivo frente aos métodos propostos na literatura para solução do problema. Como possíveis trabalhos futuros, poderia-se testar a fixação das variáveis para valores igual a um. Além disso, poderia-se testar a inclusão da restrição que foi retirada e resolvê-la através do método de geração de colunas.

Tabela 4: Comparativo entre o algoritmo *Branch-and-Bound* **BB** (Marti, R. et al., 2010) e o algoritmo **DBR** proposto para as instâncias **SOM** por Silva, G. et al. (2004),considerando um tempo limite de execução igual a uma hora (3600s) como critério de parada

			BB		DBR					BB		DBR	
Instância	**n**	**m**	**T(s)**	**GAP(%)**	**T(s)**	**GAP(%)**	**Instância**	**n**	**m**	**T(s)**	**GAP(%)**	**T(s)**	**GAP(%)**
1	25	2			0,11	0,00	26	100	30			3600	14,84
2	25	2			0,11	0,00	27	100	30			3600	15,16
3	25	2			0,11	0,00	28	100	30			3600	13,43
4	25	2			0,10	0,00	29	100	30			3600	14,87
5	25	2			0,12	0,00	30	100	30			3600	15,81
		Média	0,00	0,00	0,11	0,00			Média	3600	31,70	3600	14,82
6	25	7			0,43	0,00	31	125	12			3600	15,61
7	25	7			0,50	0,00	32	125	12			3600	14,29
8	25	7			0,46	0,00	33	125	12			3600	18,51
9	25	7			1,39	0,00	34	125	12			3600	15,80
10	25	7			0,45	0,00	35	125	12			3600	15,99
		Média	0,00	0,00	0,65	0,00			Média	2678,90	0,00	3600	16,04
11	50	5			1,94	0,00	36	125	37			3600	17,96
12	50	5			1,61	0,00	37	125	37			3600	19,04
13	50	5			1,78	0,00	38	125	37			3600	18,14
14	50	5			1,75	0,00	39	125	37			3600	18,92
15	50	5			1,79	0,00	40	125	37			3600	17,98
		Média	0,00	0,00	1,77	0,00			Média	3600	34,60	3600	18,41
16	50	15			481,58	0,00	41	150	15			3600	24,43
17	50	15			151,38	0,00	42	150	15			3600	23,35
18	50	15			838,01	0,00	43	150	15			3600	22,25
19	50	15			741,52	0,00	44	150	15			3600	21,75
20	50	15			194,46	0,00	45	150	15			3600	23,35
		Média	22,80	0,00	481,39	0,00			Média	3600	26,70	3600	23,03
21	100	10			2449,63	0,00	46	150	45			3600	26,30
22	100	10			2202,45	0,00	47	150	45			3600	25,78
23	100	10			1022,73	0,00	48	150	45			3600	26,84
24	100	10			1227,96	0,00	49	150	45			3600	27,60
25	100	10			1514,70	0,00	50	150	45			3600	26,18
		Média	38,30	0,00	1683,49	0,00			Média	3600	35,60	3600	26,54

Tabela 5: Tempo computacional gasto para o algoritmo **DBR** proposto atingir a otimalidade das instâncias **GKD** propostas por Marti, R. et al. (2010), sem tempo limite como critério de parada

Instância	**n**	**m**	**T(s)**
27	100	30	6607,31
36	125	37	89395.48
38	125	37	9360.50
39	125	37	17157.96
40	125	37	5325.80
43	150	15	4618.82
46	150	45	35652.98
47	150	45	15123.47
48	150	45	94727.12
49	150	45	11164.32
50	150	45	69379.51

Referências

Adams,W. P., & Sherali, H. D. (1986). *A tight linearization and an algorithm for zeroone quadratic programming problems*. *Management Science*, *32*(10), 1274-1290.

Adams,W. P., & Sherali, H. D. (1990). *Linearization strategies for a class of zero-one mixed integer programming problems*. *Operations Research*, *38*(2), 217-226.

Andrade, P., Plastino, A., Ochi, L., & Martins, S. (2003). *Grasp for the maximum diversity problem*. *In Proceedings of MIC*, 25-28.

Aringhieri, R., Cordone, R., & Melzani, Y. (2008). *Tabu search vs. grasp for the maximum diversity problem. 4OR: A Quarterly Journal of Operations Research.*, *1*(6), 45-60.

Cox, T. (1993). *Cultural diversity in organizations: theory, research and practice. Technical report, University of Washington, Department of Management Science.*.

Duarte, A., & Marti, R. (2007). *Tabu search and grasp for the maximum diversity problem. European Journal of Operational Research*, *178*(46), 71-84.

Erkut, E., & Neuman, S. (1991). *Comparison of four models for dispersing facilities. INFOR Canadian Journal of Operational Research and Information Processing,*, *29*, 68-86.

Fernandez, J. (1991). *Managing a diverse work force. Lexington Books, Lexington, MA.*.

Ghosh, J. (1996). *Computational aspects of the maximum diversity problem. Operations Research Letters.*, *19*, 175-191.

Glover, F., Hersh, G., & McMillian, C. (1977). *Selecting subset of maximum diversity. Technical report, University of Colorado, Boulder.*.

Glover, F., Kuo., C., & Dhir, K. (1998). *Heuristic algorithms for the maximum diversity problem. Journal of Information and Optimization Sciences*, *1*(19), 109-132.

Kuo, C., Glover, F., & Dhir, K. S. (1993). *An application of tabu search heuristic for the maximum edge-weighted subgraph problem. Decision Sciences.*, *24*, 1171-1185.

Marti, R., Gallego, M., & Duarte, A. (2010). *A branch and bound algorithm for the maximum diversity problem. European Journal of Operational Research.*, *1*(200), 36-44.

McDaniel, D., & Devine, M. (1977). *A modified benders partitioning algorithm for mixed integer programming. Management Science,*, *3*(24), 312-319.

Palubeckis, G. (2007). *Iterated tabu search for the maximum diversity problem. Applied Mathematics and Computation,*, *189*, 371-383.

Silva, G., Andrade, M., Ochi, L., Martins, S., & Plastino, A. (2007). *New heuristics for the maximum diversity problem. Lecture Notes in Computer Science.*, *4*(13), 315-336.

Silva, G., Ochi, L., & Martins, S. (2004). *Experimental comparison of greedy randomized adaptive search procedures for the maximum diversity problem. Lecture Notes in Computer Science.*, *3059*, 498-512.