

ABORDAGEM HEURÍSTICA PARA O PROBLEMA DA DIVERSIDADE MÁXIMA DE GRUPOS

Joelma Mayara da Silva

Departamento de Estatística, Universidade Federal de Pernambuco – UFPE
Cidade Universitária, 50740-540, Recife, PE
jmds1@de.ufpe.br

Geiza Cristina da Silva (*)

Departamento de Estatística, Universidade Federal de Pernambuco – UFPE
Cidade Universitária, 50740-540, Recife, PE
geiza@de.ufpe.br

RESUMO

O Problema da Diversidade Máxima de Grupos – PDMG (*Maximally Diverse Grouping Problem* – MDGP), da área de Otimização Combinatória, tem por objetivo particionar um conjunto de elementos em subconjuntos menores, chamados *grupos*, de maneira que a *diversidade* entre os elementos em cada grupo seja a maior possível. Neste trabalho são apresentadas metodologias baseadas na metaheurística GRASP (*Greedy Randomized Adaptive Search Procedure*), além de um algoritmo híbrido de GRASP e VND (*Variable Neighborhood Descent*) para gerar soluções para o problema. Os resultados computacionais mostram que este método híbrido tem maior capacidade para escapar de ótimos locais, visto que em média seus resultados são melhores do que o GRASP puro, sobretudo em instâncias de médio e grande porte.

PALAVRAS CHAVE. Problema de k -partição. Problema de Partição Equilibrada. Metaheurísticas

Área principal. MH – Metaheurísticas

ABSTRACT

The Maximally Diverse Grouping Problem (MDGP) of Combinatorial Optimization area aims to partition a set of elements into smaller subsets, called groups, so that the diversity of the elements in each group is the largest possible. In this work are presented methodologies to generate solutions to the problem, based on GRASP (Greedy Randomized Adaptive Search Procedure) as well as a hybrid algorithm GRASP and VND (Variable Neighborhood Descent). The computational results show that this hybrid method is better able to escape from local optima, since on average the results are better than pure GRASP, especially in medium and large instances.

KEYWORDS. k - partition Problem. Equitable Partition Problem. Metaheuristics.

Main area. Metaheuristics

(*) A autora agradece à FACEPE pelo financiamento parcial da pesquisa (APQ-0446-3.08/12).

1. Introdução

O Problema da Diversidade Máxima de Grupos – PDMG (*Maximally Diverse Grouping Problem* – MDGP), da área de Otimização Combinatória, tem por objetivo particionar um conjunto de elementos em subconjuntos menores, chamados *grupos*, de maneira que a *diversidade* entre os elementos em cada grupo seja a maior possível. Em Feo et al. (1992) foi provado que o PDMG é NP-hard. De maneira geral, na prática, os problemas de otimização desta natureza são caracterizados por grandes espaços de soluções dificultando, com isso, a obtenção de uma solução de custo ótimo através de um algoritmo exato em um tempo computacional suportável. Por este motivo, métodos que gerem soluções aproximadas são cada vez mais aplicados na área.

Um algoritmo aproximativo ou heurístico é um método de obtenção de soluções para um determinado problema que não garante que as soluções obtidas sejam necessariamente ótimas, mas que tenham proximidade do ótimo em tempo de computação viável. Dentre os mais diversos métodos heurísticos encontram-se as metaheurísticas que têm se mostrado bastante eficientes no que se trata dos fatores de qualidade da solução e tempo de computação.

Neste trabalho são apresentadas metodologias para gerar soluções para o problema, baseadas na metaheurística GRASP (*Greedy Randomized Adaptive Search Procedure*) além de um algoritmo híbrido de GRASP e VND (*Variable Neighborhood Descent*).

O restante deste artigo está organizado da seguinte forma: A seção 2 apresenta os principais aspectos do PDMG. Na seção 3 são apresentados os algoritmos propostos e na seção 4, os resultados computacionais alcançados. A seção 5 encerra o trabalho trazendo algumas conclusões e sugestões para trabalhos futuros.

2. O Problema da Diversidade Máxima de Grupos (PDMG)

O PDMG é o problema de, a partir de um conjunto N de cardinalidade n , selecionar os elementos para compor m grupos, $m \leq n$, de modo que a diversidade entre os elementos em cada grupo seja maximizada.

O índice de diversidade representa o grau de diferença existente entre os atributos dos elementos i e j , que pode ser entendida como a distância entre eles e deve ser calculada usando-se uma dentre as métricas de distâncias existentes. De acordo com a área de aplicação, uma métrica pode refletir melhor o problema que outra (Kochenberger e Glover, 1999). Desta forma, para definir o problema considera-se o conjunto de índices $N = \{1, 2, \dots, n\}$, e:

- t , o número de atributos que caracterizam os elementos de N ;
- a_{ik} , o valor do atributo k do elemento i , $k = \{1, 2, \dots, t\}$, $i \in N$;
- $D = (d_{ij})$ uma matriz simétrica $n \times n$, onde cada d_{ij} corresponde ao índice de diversidade entre dois elementos distintos i e j , que possuem os respectivos valores de atributos $(a_{i1}, a_{i2}, \dots, a_{it})$ e $(a_{j1}, a_{j2}, \dots, a_{jt})$, $(i, j \in N, i \neq j)$.

Existem duas variantes do PDMG: Uma em que todos os grupos têm o mesmo número de elementos, isto é, $k = n/m$ elementos. A outra, permite tamanhos variáveis de grupos. Isto é, o número c_g de elementos de um grupo g é definido no intervalo $[a_g, b_g]$, com $a_g \leq b_g$, $g = 1, 2, \dots, m$. Ambas as variantes podem ser formuladas como um problema de programação inteira considerando-se x_{ig} ($i = 1, 2, \dots, n$ e $g = 1, 2, \dots, m$) as variáveis binárias que se tornam 1, se e somente se o elemento i pertence ao grupo g e 0, caso contrário.

Maximizar

$$\sum_{g=1}^m \sum_{i=1}^{n-1} \sum_{j=i+1}^n d_{ij} x_{ig} x_{jg} \quad (1)$$

sujeito a:

$$\sum_{g=1}^m x_{ig} = 1, i = 1, 2, \dots, n, \quad (2)$$

$$\sum_{i=1}^n x_{ig} \geq a_g, g = 1, 2, \dots, m, \quad (3)$$

$$\sum_{i=1}^n x_{ig} \leq b_g, g = 1, 2, \dots, m, \quad (4)$$

$$x_{ig} \in \{0,1\}, i = 1, 2, \dots, n, g = 1, 2, \dots, m. \quad (5)$$

A formulação é um modelo não linear devido a função objetivo quadrática (1) e tem por meta maximizar a soma das diversidades entre os elementos de cada grupo. As restrições 2 exigem que cada elemento i seja incluído em exatamente um grupo g . As restrições 3 e 4 asseguram que o grupo g contenha ao menos a_g e não mais que b_g elementos. As restrições 5 determinam que as variáveis de decisão são binárias.

O PDMG pode ser encontrado na literatura com outros nomes, como problema de k -partição (k -partition problem, Feo et al., 1992) ou Problema de Partição Equilibrada (*equitable partition problem*, O'Briene Mingers, 1995). Além disso, o Problema da Diversidade Máxima (PDM) (Kuo et al., 1993, Silva et al., 2007) é o caso particular mais explorado na literatura em que o número de grupos é sempre igual a 1.

Várias são as aplicações potenciais do PDMG levando-se em consideração que em muitas situações práticas é desejável criar grupos diversos. Algumas encontradas na literatura são: associação de estudantes a grupos de estudos ou de projetos (Weitz e Lakshminarayanan, 1997; Weitz e Lakshminarayanan, 1998; Weitz, Jelassi, 1992); criação de grupos de revisores para publicações científicas ou para avaliação de projetos em agências de fomento à ciência (Hettich e Pazzani, 2006); projetos de circuitos integrados (Feo e Khellaf, 1990).

Diversas abordagens heurísticas e metaheurísticas têm sido propostas para obter soluções de alta qualidade para o PDMG.

Fan et al. (2011) abordaram o PDMG através de um algoritmo híbrido baseado em algoritmos genéticos. Esta abordagem mostrou-se satisfatória encontrando-se bons resultados para instâncias de tamanho máximo $n = 240$.

Gallego et al. (2013) propuseram uma nova heurística utilizando mecanismos de busca, com base na metodologia busca tabu para o PDMG, incluindo uma *estratégia de oscilação*, que permitiu o cruzamento de caminhos viáveis de busca. Foram avaliados procedimentos de construção e de busca para configurar um procedimento de solução que quando comparado ao estado da arte para o MDGP. Extensos experimentos computacionais com médias e grandes instâncias foram apresentados para demonstrar as vantagens do método de solução que inclui estratégia de oscilação.

Palubeckis et al. (2011) propuseram três heurísticas para o PDMG, sendo elas: *Multistart simulated annealing* (MSA), *Hybrid steady-state genetic algorithm* (HGA) e *Variable Neighbourhood search* (VNS). O desempenho das heurísticas foi comparado computacionalmente através de dois grupos de instâncias: o primeiro, instâncias do Problema da Diversidade Máxima, caso particular do PDMG e o outro, com instâncias de até 2000 elementos. Dentre as três abordagens, a que se mostra mais eficiente é o HGA.

Urošević (2014) desenvolveu uma heurística híbrida baseada em VNS onde o processo de busca utilizado para melhoria de soluções baseou-se em um algoritmo VND composto por duas

vizinhanças. Os resultados apresentados mostram que a metodologia proposta é 0,5% melhor que os resultados obtidos com o VNS clássico.

Rodriguez et al. (2013) apresentam um algoritmo *Artificial Bee Colony* (ABC) que é uma metodologia baseada no comportamento alimentar das abelhas. A eficiência do novo algoritmo foi comprovada para instâncias de até 2000 elementos e, comparado aos seus antecessores, este algoritmo apresentou as melhores soluções até o presente momento.

3. Metodologia Proposta

Esta seção apresenta os algoritmos propostos neste trabalho composta de dois métodos de gerar soluções iniciais nos moldes da construção GRASP, três estruturas de vizinhança utilizadas em seis diferentes buscas locais. Combinados, algoritmos de construção e de busca local são utilizados para gerar 12 diferentes métodos GRASP e um algoritmo híbrido de GRASP e VND.

3.1 Algoritmos de Construção:

- Algoritmo de Construção 1: Numa primeira fase atribui-se um elemento a cada um dos grupos da seguinte forma: calcula-se a soma das distâncias entre um elemento e todos os outros, e ordena-se os elementos por ordem crescente desta soma. Cada um dos m primeiros elementos é associado a um dos m grupos escolhido aleatoriamente, encerrando-se a primeira etapa. A segunda fase do procedimento completará os grupos até que cada qual tenha o número mínimo de elementos a_g : A cada iteração do método, para cada elemento i (ainda não associado a grupo) e cada grupo g (com número de elementos já associados menor do que a_g), calcula-se a média com a soma das distâncias entre i e os elementos contidos em g e a quantidade de elementos em g . Uma lista de candidatos (LC) é formada com esses elementos ordenados em ordem crescente da média calculada e, a partir desta, os $\alpha \times (\text{tamanho da LC})$ primeiros elementos são considerados para a escolha aleatória que levará à associação de um elemento a um grupo, α é um parâmetro do método GRASP. Estas iterações se repetem até que todos os grupos tenham a_g elementos. Uma observação a ser feita nesta fase é que, é possível que a LC tenha muitos elementos (da ordem de $n \times m$) fazendo com que, em instâncias com grandes valores de m e n , o cálculo de todas as possibilidades torne o método muito lento. Por esse motivo, a metodologia prevê um número máximo de elementos ($\text{MAX_CAND}/m$) por grupo a serem testados, escolhidos aleatoriamente. O procedimento termina aqui para a variante do problema em que os grupos têm o mesmo tamanho.
Para a outra variante, quando os grupos têm tamanhos diferentes, ocorre ainda uma terceira fase, em que cada grupo ficará com o número de elementos associados entre a_g e b_g . Para isto, o critério de construção da segunda fase é repetido com a alteração de restringir a inserção a grupos em que o número de elementos não ultrapasse o número máximo permitido b_g .
- Construção 2: Difere da Construção 1 apenas por sua primeira fase, com a escolha do primeiro elemento de cada grupo feita de forma aleatória.

3.2 Algoritmos de Busca Local

Um algoritmo de busca local tem por objetivo explorar a vizinhança da solução inicial a fim de buscar uma solução melhor. A busca por melhores soluções pode seguir estratégias diferentes entre si. Neste trabalho foram utilizadas duas estratégias:

- Primeiro de melhora: Neste tipo de estratégia explora-se a vizinhança de uma solução através de movimentos realizados na solução corrente. A cada movimento realizado, sempre que há melhora na solução corrente, a nova solução obtida é aceita e passa a ser a solução corrente. O processo é repetido até que, realizados todos os movimentos em uma solução, nenhuma melhoria possa ser encontrada.

- Melhor vizinho: Neste caso, a partir de uma solução corrente, são testados todos os movimentos possíveis e o melhor movimento encontrado é realizado gerando uma nova solução corrente. O processo é repetido até que, após serem realizados todos os possíveis movimentos, nenhuma melhoria seja alcançada.

Foram implementadas três diferentes estruturas de vizinhanças, propostas inicialmente por Urošević (2014):

- Troca de 2 elementos: Para cada elemento i e j associados a dois diferentes grupos g_i e g_j , respectivamente, a estrutura de vizinhança consiste de todas as possíveis trocas de grupos entre i e j .
- Troca de 3 elementos: Para três elementos i , j e k , pertencentes a três diferentes grupos g_i , g_j e g_k , respectivamente, a estrutura de vizinhança consiste nas seguintes trocas: i de g_i para g_j , j de g_j para g_k e k de g_k para g_i .
- Realocação: Nesta estrutura de vizinhança, um elemento i é retirado de um grupo g_i e realocado em um grupo g_j desde que a solução mantenha-se viável, ou seja, após a execução do movimento o número de elementos em g_i permaneça maior ou igual a a_g e o número de elementos em g_j permaneça menor ou igual a b_g . Mais especificamente, um movimento de realocação é realizado para cada elemento i tal que $|g_i| \geq a_g$ e para cada grupo g_j (g_i diferente de g_j) tal que $|g_j| \leq b_g$.

3.3 Heurísticas GRASP Propostas

A metaheurística GRASP proposta inicialmente por Feo e Resende (1995) tem se mostrado bastante eficiente para resolver diversos problemas de otimização combinatória (Silva et al., 2007; Resende e Ribeiro, 2014). É um método iterativo composto por duas fases: construção de uma solução e busca local.

A fase de construção tem algumas propriedades particulares, pois, é ao mesmo tempo iterativa, gulosa, randômica e adaptativa. Ela é iterativa porque constrói uma solução elemento a elemento e é adaptativa, pois a escolha do próximo elemento da solução parcial é influenciada pelas escolhas anteriores. Para a seleção do próximo elemento da solução parcial, a princípio todos os candidatos são considerados, contudo o número de candidatos em muitos casos pode ser elevado, por isso, geralmente, é considerada apenas uma lista restrita de candidatos (LRC). O tamanho da LRC é $p = 1 + \alpha (a - 1)$, onde a é o número total de elementos candidatos e α é um parâmetro de entrada do método. A partir de uma LRC, seleciona-se um elemento deste conjunto aleatoriamente e não necessariamente o melhor. Esta escolha aleatória permite que este procedimento possa ser usado várias vezes para obter soluções distintas.

Pela forma como são construídas as soluções na etapa inicial, estas não representam na maioria dos casos um ótimo local, daí ser fortemente recomendada uma etapa de busca local. As iterações GRASP são totalmente independentes. O critério de parada do GRASP geralmente é o número máximo de iterações e a solução final GRASP é a melhor solução obtida ao fim da execução.

Os algoritmos descritos anteriormente foram combinados de maneira que cada construção, par de estratégia de busca e tipo de vizinhança utilizada gerou um algoritmo do tipo GRASP.

Tabela 1 - Algoritmos GRASP Propostos

Algoritmo de Construção	Estrutura de Vizinhança	Estratégia de Busca	Algoritmo GRASP Proposto
Construção 1	Troca de 2 elementos	Primeiro de Melhora	G1
		Melhor vizinho	G2
	Troca de 3 elementos	Primeiro de Melhora	G3
		Melhor vizinho	G4
	Realocação	Primeiro de Melhora	G5
		Melhor vizinho	G6
Construção 2	Troca de 2 elementos	Primeiro de Melhora	G7
		Melhor vizinho	G8
	Troca de 3 elementos	Primeiro de Melhora	G9
		Melhor vizinho	G10
	Realocação	Primeiro de Melhora	G11
		Melhor vizinho	G12

3.4 Heurística GRASP+VND Proposta

A metaheurística VND tem o objetivo de escapar de ótimos locais, pela exploração sistemática de diferentes vizinhanças em uma solução. Considera-se $kmax$ estruturas de vizinhanças $N^{(k)}$ ($k = 1, \dots, kmax$) e uma solução corrente s . O procedimento é iniciado com $k = 1$ e um *loop* é executado enquanto $k \neq kmax$. Para cada k , os possíveis movimentos são avaliados. Se houver melhora, a solução obtida passa a ser a solução corrente e k é reiniciado em 1. Quando toda uma vizinhança k é explorada e não pode ser encontrado nenhum movimento que melhore a solução corrente, k é incrementado. O pseudocódigo do procedimento é apresentado na figura 1.

Figura 1 – Pseudocódigo método VND

```

Início VND
S0 = solução inicial
S ← S0
k ← 1
Enquanto (k ≤ kmax) faça
  Melhor vizinho s' ∈ N(k)
  se f(s') > f(s)
    s ← s'
    k ← 1
  else
    k ← k + 1
  fim se
fim enquanto
return s
Fim VND
  
```

Neste trabalho propomos um algoritmo híbrido GRASP e VND da seguinte forma: os algoritmos de busca local utilizados nos algoritmos GRASP são substituídos pelo método de busca em vizinhança variável, o VND. São utilizadas as três estruturas de vizinhanças descritas na seção 3.2: Realocação, Troca de 3 elementos e Troca de 2 elementos, nesta ordem.

Utilizando-se dos algoritmos de construção descritos na seção 3.1, geramos dois algoritmos híbridos: GVND1 e GVND2.

4 Resultados Computacionais

Esta seção descreve os experimentos computacionais realizados para testar a eficiência dos procedimentos discutidos anteriormente. Todos os métodos foram implementados em C. Todos

os testes foram conduzidos em um processador Intel Core i7-4770 8 Threads com 16 Gb de RAM e Ubuntu 14.04 64 bits.

Os experimentos foram realizados sobre 480 instâncias do problema. Este conjunto de instâncias, referido como MDGPLIB, é disponibilizado em <http://www.opticom.es/mdgp> e dividido em três subconjuntos:

RanReal: 160 instâncias compostas de matrizes $N \times N$ em que os valores das distâncias são números reais gerados usando uma distribuição uniforme $U(0,100)$. O número de elementos n , o número de grupos m e os limites de cada grupo a_g e b_g são mostrados na tabela 2. São 20 instâncias para cada combinação de parâmetros, 10 instâncias para o caso em que todos os grupos devem ter o mesmo tamanho e 10 instâncias de grupos de tamanhos diferentes. No primeiro caso, $a_g = b_g = n/m$. Já no segundo caso, os limites de cada grupo (a_g e b_g) foram gerados aleatoriamente em um intervalo pré-definido $[a_g^{min}, a_g^{max}]$ e $[b_g^{min}, b_g^{max}]$, respectivamente. O conjunto de instâncias com n de 10 a 240 foi introduzido por Fan et al. (2011). Gallego et al. (2013) geraram instâncias maiores, de 480 e 960 elementos.

Tabela 2 – Sumário dos parâmetros usados para gerar as instâncias

n	m	$a_g = b_g$	a_g^{min}	a_g^{max}	b_g^{min}	b_g^{max}
10	2	5	3	5	5	7
12	4	3	2	3	3	5
30	5	6	5	6	6	10
60	6	10	7	10	10	14
120	10	12	8	12	12	16
240	12	20	15	20	20	25
480	20	24	18	24	24	30
960	24	40	32	40	40	48

RanInt: Tem a mesma estrutura e tamanho de *RanReal* mas as distâncias são inteiros gerados por uma distribuição uniforme $U(0,100)$.

Geo: Segue a mesma estrutura e tamanho dos outros dois conjuntos, entretanto, as distâncias são calculadas como distância Euclidiana entre pares de pontos com coordenadas aleatoriamente geradas no intervalo $[0,10]$. O número de coordenadas para cada instância é aleatoriamente gerado no intervalo entre 2 e 21. Glover et al. (1998) propôs este gerador para o PDM.

Nos experimentos apresentados a seguir foram computados para cada instância a melhor solução obtida durante a execução de cada método em questão. Em seguida, foram calculados os desvios percentuais entre a melhor solução encontrada por cada método e o melhor valor levando em consideração todos os métodos.

Todos os métodos utilizam o mesmo critério de parada, um tempo limite, que varia de acordo com o tamanho do problema, como descrito na tabela 3.

Tabela 3 – Tempos limites de execução

n	Segundos
≤ 60	1
120	3
240	20
480	120
960	600

Os parâmetros utilizados nos algoritmos de construção foram estabelecidos através de testes preliminares, como: $\alpha = 0,1$, MAX_CAND = 1500.

A tabela 4 faz a comparação entre os algoritmos propostos levando-se em consideração todas as instâncias: G1 a G12. Na tabela são apresentados, na primeira linha o algoritmo em

questão, na segunda linha, o desvio percentual e na última linha, o número de melhores soluções alcançadas por cada algoritmo, considerando o total de 480 instâncias.

Tabela 4 – Comparação entre os métodos GRASP

Alg.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12
Desvio (%)	3,24	3,36	3,47	8,34	6,90	8,20	3,22	3,33	3,39	8,08	6,65	7,61
#Melhores	177	168	117	23	40	24	176	171	125	36	52	36

Considerando que os seis primeiros métodos utilizam a Construção 1 e os demais a Construção 2, pode-se observar que o algoritmo Construção 2 apresenta resultados ligeiramente melhores. Pode-se verificar também que os algoritmos de busca que utilizam a estratégia primeiro de melhora têm melhores resultados do que a do melhor vizinho. Quanto às estruturas de vizinhança podem ser observadas grandes diferenças de desempenho. A estrutura que propõe a troca de 2 elementos se mostra mais eficiente que as demais. Os algoritmos G3, G4 e G9 e G10 apresentam diferenças importantes de desvio que tanto pode ser devida à mudança entre estratégia de busca como à estrutura de vizinhança ou a combinação destes dois fatores. Em negrito é destacado o algoritmo que teve melhores resultados, G7. Além disso, G1 apresenta resultados muito próximos.

Na tabela 5 é apresentada a comparação de desempenho entre os dois melhores algoritmos GRASP e os algoritmos híbridos GRASP e VND.

Tabela 5 – Comparação entre os métodos GRASP e híbridos GRASP e VND

Alg.	G1	G7	GVND1	GVND2
Desvio (%)	3,23	3,22	2,51	0,80
#Melhores	178	179	328	298

É possível notar uma melhora significativa nos resultados obtidos pelos métodos que utilizam como busca local a metodologia VND, com as três vizinhanças trabalhando em conjunto, ante àqueles que exploram uma única estrutura de vizinhança.

A tabela 6 mostra uma comparação entre variantes do PDMG, isto é, entre instâncias que têm grupos de tamanhos diferentes e aquelas que têm o mesmo tamanho para os grupos. Cada tipo é constituído de 240 instâncias. Vê-se que o comportamento dos algoritmos permanece o mesmo com uma ligeira melhora no caso de instâncias do segundo tipo.

Tabela 6 – Comparação entre os métodos GRASP e híbridos GRASP e VND por tipo de problema

Tipo de Problema	Alg.	G1	G7	GVND1	GVND2
Tamanhos de grupos Diferentes	Desvio (%)	3,88	3,84	2,57	0,97
	#Melhores	72	73	179	136
Tamanhos de grupos Iguais	Desvio (%)	2,60	2,59	2,45	0,63
	#Melhores	106	106	149	162

A tabela 7 mostra os resultados obtidos pelos métodos considerando a subdivisão das instâncias em *RanReal*, *RanInt*, *Geo*, cada qual com 160 instâncias.

Tabela 7 – Comparação entre os métodos GRASP e híbridos GRASP e VND por tipo de instância

Tipo de Instância	Alg.	G1	G7	GVND1	GVND2
<i>RanReal</i>	Desvio (%)	1,43	1,43	0,59	0,33
	#Melhores	59	60	115	102
<i>RanInt</i>	Desvio (%)	1,71	1,70	0,85	0,10
	#Melhores	55	56	118	99
<i>Geo</i>	Desvio (%)	6,56	6,53	6,10	1,96
	#Melhores	64	63	95	97

É interessante ver que os métodos têm um comportamento similar para os dois primeiros tipos de instâncias mas isso não acontece com as instâncias *Geo*. Neste caso em particular o algoritmo GVND1 tem seu desvio médio muito mais próximos dos métodos GRASP puros do que de GVND2. Este por sua vez tem o melhor desempenho, podendo-se supor que para as características destas instâncias o algoritmo de construção com maior aleatoriedade proporcionou ao VND maior poder para escapar de ótimos locais.

Por fim, a tabela 8 mostra a comparação por tamanho de instâncias como em Gallego et al (2013).

Tabela 8 – Comparação entre os métodos GRASP e híbridos GRASP e VND por tamanho de instância

Tamanho de Instância	Alg.	G1	G7	GVND1	GVND2
$n \leq 60$ (240 instâncias)	Desvio (%)	5,20	5,18	4,99	1,50
	#Melhores	173	174	203	193
$n = 120$ e $n = 240$ (120 instâncias)	Desvio (%)	1,39	1,40	0,05	0,13
	#Melhores	4	6	60	50
$n = 480$ e $n = 960$ (120 instâncias)	Desvio (%)	1,15	1,12	0,03	0,05
	#Melhores	0	0	65	55

É interessante notar que conforme aumentam as instâncias, os algoritmos GRASP puros sofrem uma queda de desempenho. Enquanto os algoritmos GRASP e VND permanecem liderando com as melhores soluções. Há ainda de se destacar que o algoritmo GVND1 apresenta uma melhor performance do que GVND2 para instâncias de maiores dimensões.

4 Conclusões e sugestões para trabalhos futuros

Neste trabalho foram apresentados novos métodos baseados em GRASP para gerar soluções viáveis para o PDMG. Além disso, foi proposta uma versão que utiliza o algoritmo GRASP em conjunto com uma busca em vizinhança variável, método conhecido como VND. Para isto foram utilizadas três estruturas de vizinhança diferentes. Foi mostrado que este método híbrido tem maior capacidade para escapar de ótimos locais, visto que em média seus resultados são melhores do que o GRASP puro, sobretudo em instâncias de médio e grande porte.

Como trabalhos futuros propõe-se a comparação dos métodos VND com os algoritmos da literatura.

Referências

- Fan, Z.; Chen, Y.; Ma, J.; Zeng, S.** (2011), A hybrid genetic algorithmic approach to the maximally diverse grouping problem, *Journal of the Operational Research Society* 62, 92–99.
- Feo, T.; Goldschmidt, O.; Khellaf, M.** (1992), One-half approximation algorithms for the k-partition problem, *Operations Research* 40, 170–173.
- Feo, T.; Khellaf, M.** (1990), A class of bounded approximation algorithms for graph partitioning, *Networks* 20, 181–195.

Gallego, M.; Laguna, M.; Martí, R.; Duarte, A. (2013), Tabu search with strategic oscillation for the maximally diverse grouping problem, *Journal of Operational Research Society* 64, 724-734.

Glover, Fred; Ching-Chung Kuo; Krishna S. Dhir (1998), Heuristic algorithms for the maximum diversity problem. *Journal of Information and Optimization Sciences* 19.1: 109-132.

Hettich, S.; Pazzani, M.J. (2006), Mining for proposal reviewers: lessons learned at the national science foundation, in: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 862-871.

Kochenberger, G.; Glover, F., Diversity data mining, *Relatório Técnico*, The University of Mississippi, 1999.

Kuo, C.-C.; Glover, F.; Dhir, K., (1993), Analyzing and modeling the maximum diversity problem by zero-one programming. *Decision Science*, 24, 1171-1185.

O'Brien, F.; Mingers, J., **The equitable partitioning problem: a heuristic algorithm applied to the allocation of university student accommodation**, Warwick Business School, Relatório Técnico, 1995.

Palubeckis, G.; Karciauskas, A.R.E. (2011), Comparative performance of three metaheuristic approaches for the maximally diverse grouping problem, *Information Technology and Control* 40, 277-285.

Resende, M.G.C.; Ribeiro, C.C., GRASP: Greedy Randomized Adaptive Search Procedures, *Search Methodologies*, 2nd edition, E.K. Burke and G. Kendall (Eds.), Chapter 11, pp. 287-310, Springer, 2014.

Rodriguez, F. J., Lozano, M., García-Martínez, C., & González-Barrera, J. D. (2013), An artificial bee colony algorithm for the maximally diverse grouping problem. *Information Sciences*, 230, 183-196.

Sandoya, F.; Aceves, R. (2013), Grasp and Path Relinking to Solve the Problem of Selecting Efficient Work Teams, *Recent Advances on Meta-Heuristics and Their Application to Real Scenarios*, Dr. Javier Del Ser Lorente (Ed.), ISBN: 978-953-51-0913-6, InTech, DOI: 10.5772/53700.

Silva, G. C.; Andrade, M.; Ochi, L.S.; Martins, S.L.; Plastino, A. (2007), New heuristics for the maximum diversity problem, *Journal of Heuristics*, 13, 315-336.

Urošević, D (2014), Variable neighborhood search for maximum diverse grouping problem. *Yugosl. J. Oper. Res.* 24, 21-33.

Weitz, R.; Lakshminarayanan, S. (1997), An empirical comparison of heuristic and graph theoretic methods for creating maximally diverse groups, VLSI design and exam scheduling, *Omega* 25, 473-482.

Weitz, R.; Lakshminarayanan, S. (1998), An empirical comparison of heuristic methods for creating maximally diverse groups, *Journal of Operation Research Society* 49, 635-646.

Weitz, R.R.; Jelassi, M.T. (1992), Assigning students to groups: a multi-criteria decision support system approach, *Decision Sciences* 23, 746-757.