



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
DE COMPUTAÇÃO



Formulações e algoritmos para o problema das p -medianas heterogêneo livre de penalidade

Éverton Santi

Orientador: Prof. Dr. Daniel Aloise

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e de Computação da UFRN (área de concentração: Engenharia de Computação) como parte dos requisitos para obtenção do título de Doutor em Engenharia de Computação.

Número de ordem PPgEE: D128
Natal, RN, novembro de 2014

UFRN / Biblioteca Central Zila Mamede

Catálogo da Publicação na Fonte.

Santi, Éverton

Formulações e algoritmos para o problema das p-medianas heterogêneo livre de penalidade / Éverton Santi. - Natal, RN, 2014.

108 f. : il.

Orientador: Prof. Dr. Daniel Aloise.

Tese (Doutorado) - Universidade Federal do Rio Grande do Norte. Centro de Tecnologia. Programa de Pós-Graduação em Engenharia Elétrica e de Computação.

1. Otimização Combinatória - Tese. 2. Problema das p-Mediana Heterogêneo - Tese. 3. Segmentação de Consumidores - Tese. I. Aloise, Daniel. II. Universidade Federal do Rio Grande do Norte. III. Título.

RN/UF/BCZM

CDU 519.863

Formulações e algoritmos para o problema das p -medianas heterogêneo livre de penalidade

Éverton Santi

Tese de Doutorado aprovada em 14 de novembro de 2014 pela banca examinadora composta pelos seguintes membros:




Prof. Dr. Daniel Aloise (orientador) DCA/UFRN



Prof. Dr. Simon Blanchard Georgetown University



Prof. Dr. Adrião Duarte Doria Neto DCA/UFRN



Prof.ª Dr.ª Caroline Thennecy de Medeiros Rocha ECT/UFRN



Prof. Dr. Sebastián Alberto Urrutia UFMG

Para Maria, Elido e Adriana.

Agradecimentos

Ao meu orientador, Prof. Daniel, pela oportunidade, conhecimentos e amizade compartilhados.

Ao Prof. Simon, por suas importantes contribuições neste trabalho.

Ao amigo e Prof. Luciano, um dos grandes motivadores desta conquista.

Aos membros da banca examinadora, por sua participação efetiva na melhoria da qualidade deste trabalho.

À coordenação do PPgEE pelo suporte oferecido durante todo este processo.

Aos muitos Mestres que cruzaram meu caminho ao longo de toda uma vida.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo apoio financeiro concedido a este trabalho.

Resumo

Apresenta-se neste trabalho um novo modelo para o Problema das p -Medianas Heterogêneo (PPMH), proposto para recuperar a estrutura de categorias não-observadas presente em dados oriundos de uma tarefa de triagem, uma abordagem popular que possibilita entender a percepção heterogênea que um grupo de indivíduos tem em relação a um conjunto de produtos ou marcas. Este novo modelo é chamado Problema das p -Medianas Heterogêneo Livre de Penalidade (PPMHLP), uma versão mono-objetivo do problema original, o PPMH. O parâmetro principal do modelo PPMH é também eliminado, o fator de penalidade. Este parâmetro é responsável pela ponderação dos termos de sua função objetivo. O ajuste do fator de penalidade controla a maneira como o modelo recupera a estrutura de categorias não-observadas presente nos dados e depende de um amplo conhecimento do problema. Adicionalmente, duas formulações complementares para o PPMHLP são apresentadas, ambas problemas de programação linear inteira mista. A partir destas formulações adicionais, limitantes inferiores foram obtidos para o PPMHLP. Estes valores foram utilizados para validar um algoritmo de Busca em Vizinhança Variada (VNS), proposto para resolver o PPMHLP. Este algoritmo obteve soluções de boa qualidade para o PPMHLP, resolvendo instâncias geradas de forma artificial por meio de uma Simulação de Monte Carlo e instâncias reais, mesmo com recursos computacionais limitados. As estatísticas analisadas neste trabalho sugerem que o novo algoritmo e modelo, o PPMHLP, pode recuperar de forma mais precisa que o algoritmo e modelo original, o PPMH, a estrutura de categorias não-observadas presente nos dados, relacionada à percepção heterogênea dos indivíduos. Por fim, um exemplo de aplicação do PPMHLP é apresentado, bem como são consideradas novas possibilidades para este modelo, estendendo-o a ambientes *fuzzy*.

Palavras-chave: Problema das p -Medianas Heterogêneo, Segmentação de Consumidores, Otimização Combinatória.

Abstract

This work presents a new model for the Heterogeneous p -median Problem (HPM), proposed to recover the hidden category structures present in the data provided by a sorting task procedure, a popular approach to understand heterogeneous individual's perception of products and brands. This new model is named as the Penalty-free Heterogeneous p -median Problem (PFHPM), a single-objective version of the original problem, the HPM. The main parameter in the HPM is also eliminated, the penalty factor. It is responsible for the weighting of the objective function terms. The adjusting of this parameter controls the way that the model recovers the hidden category structures present in data, and depends on a broad knowledge of the problem. Additionally, two complementary formulations for the PFHPM are shown, both mixed integer linear programming problems. From these additional formulations lower-bounds were obtained for the PFHPM. These values were used to validate a specialized Variable Neighborhood Search (VNS) algorithm, proposed to solve the PFHPM. This algorithm provided good quality solutions for the PFHPM, solving artificial generated instances from a Monte Carlo Simulation and real data instances, even with limited computational resources. Statistical analyses presented in this work suggest that the new algorithm and model, the PFHPM, can recover more accurately the original category structures related to heterogeneous individual's perceptions than the original model and algorithm, the HPM. Finally, an illustrative application of the PFHPM is presented, as well as some insights about some new possibilities for it, extending the new model to fuzzy environments.

Keywords: Heterogeneous p -median Problem, Consumer Segmentation, Combinatorial Optimization.

Sumário

Sumário	i
Lista de Figuras	iii
Lista de Tabelas	v
Lista de Algoritmos	vii
Lista de Símbolos e Abreviaturas	ix
1 Introdução	1
1.1 O problema das p -medianas heterogêneo	5
1.2 Objetivos	10
1.3 Estrutura e organização do texto	11
2 Reformulações para o problema das p-medianas heterogêneo livre de penalidade	13
2.1 O problema das p -medianas heterogêneo livre de penalidade	14
2.2 A formulação PPMHLP1	16
2.3 A formulação PPMHLP2	20
2.4 Resultados computacionais	23
3 Um algoritmo VNS para o problema das p-medianas heterogêneo livre de penalidade	29
3.1 Heurísticas e metaheurísticas	30
3.2 O algoritmo de busca vizinhança variada	31
3.3 VNS-PPMHLP: formulação	33
3.4 VNS-PPMHLP: resultados computacionais	41
3.4.1 Análise de estabilidade e desempenho	42
3.4.2 Recuperação da informação	50
3.4.3 Sensibilidade a diferentes fatores	53

3.4.4	Exemplo de aplicação	58
4	Uma extensão do problema das p-medianas heterogêneo livre de penalidade a ambientes <i>fuzzy</i>	67
4.1	Motivação	68
4.2	Um modelo <i>fuzzy</i> para o PPMHLP	70
4.3	Exemplo de aplicação	76
4.4	Considerações	78
5	Considerações finais	81
	Referências bibliográficas	85

Lista de Figuras

1.1	Exemplo de categorização heterogênea	5
-----	--	---

Lista de Tabelas

2.1	Estrutura das instâncias simuladas	24
2.2	Resultados computacionais para as formulações PPMHLP1 e PPMHLP2 obtidos por meio do <i>solver</i> CPLEX 12.5	26
2.3	Total de variáveis de decisão para PPMHLP1 e PPMHLP2	27
3.1	VNS-PPMHLP: custos das soluções obtidas para 10 execuções	44
3.2	VNS-PPMHLP: heurística construtiva e resultados finais	48
3.3	Comparação entre resultados computacionais obtidos para as formulações PPMHLP1 e PPMHLP2 via <i>solver</i> em relação ao VNS-PPMHLP	49
3.4	Simulação de Monte Carlo: precisão dos algoritmos	55
3.5	Simulação de Monte Carlo: fatores que influenciam na recuperação da estrutura de categorias (e_{jk}^g)	56
3.6	Simulação de Monte Carlo: fatores que influenciam na recuperação da pertinência aos segmentos (p^{ig})	57
3.7	Exemplo de aplicação: 10 execuções para o VNS-PPMHLP e VNS-PPMH .	60
3.8	Exemplo de aplicação: seleção de modelo (G)	62
3.9	Exemplo de aplicação: estrutura de categorias	65
4.1	Estruturas de categorias formada pelos indivíduos 1, 2 e 3	69
4.2	Resultado do PPMHLP para o exemplo dado	70
4.3	Resultado do PPMHLP- <i>fuzzy</i> para o exemplo dado e valores de pertinên- cia $p^{(i,g)}$	77

Lista de Algoritmos

1	Estrutura de um algoritmo VNS	32
2	VNS-PPMHLP: heurística construtiva	34
3	Estrutura de um VND	37
4	VNS-PPMHLP: \mathcal{N}_3	40

Lista de Símbolos e Abreviaturas

ARI	Índice de Rand Ajustado
DP	Desvio Padrão
GVNS	Generalized Variable Neighborhood Search
PCF	Problema de <i>Clustering Fuzzy</i>
PLIM	Programação Linear Inteira Mista
PM	Problema das p -Medianas
PNLIM	Programação Não-Linear Inteira Mista
PPLIM	Problema de Programação Linear Inteira Mista
PPM	Problema das p -Medianas
PPMH	Problema das p -Medianas Heterogêneo
PPMHLP	Problema das p -Medianas Heterogêneo Livre de Penalidade
PPMHLP- <i>Fuzzy</i>	Problema das p -Medianas Heterogêneo Livre de Penalidade <i>Fuzzy</i>
PPNLIM	Problema de Programação Não-Linear Inteira Mista
SMC	Simulação de Monte Carlo
VND	Variable Neighborhood Descent
VNS-PPMH	Algoritmo de Busca em Vizinhança Variada para o Problema das p -medianas Heterogêneo
VNS-PPMHLP	Algoritmo de Busca em Vizinhança Variada para o Problema das p -Medianas Heterogêneo Livre de Penalidade

Capítulo 1

Introdução

A categorização é uma ferramenta de extrema importância desde os primórdios da humanidade e ao longo de seu desenvolvimento [Anderberg 1973], pois para que se possa aprender sobre um novo objeto ou fenômeno, buscam-se características para descrevê-lo. Desta forma, pode-se compará-lo aos objetos e fenômenos já conhecidos por meio de similaridades, padrões ou regras [Xu & Wunsch 2005].

Nos dias de hoje, o intenso avanço da tecnologia faz com que se tenham quantidades cada vez maiores de dados, tornando-se inviável analisá-los sem o uso de computadores e métodos apropriados [Liu et al. 2005]. Esta necessidade tem motivado inúmeros pesquisadores ao longo dos anos, sendo que, na literatura relacionada, a categorização está fortemente ligada à utilização de modelos e algoritmos de *clustering*.

De uma forma generalista, *clustering*, ou análise por *clusters*, pode ser entendida como a tarefa de agrupar um conjunto de objetos de forma que os objetos mais similares deste conjunto sejam colocados em um mesmo grupo, assim como objetos diferentes sejam colocados em *clusters* distintos. Do ponto de vista *de cima para baixo* ou *top-down*, considera-se que *clustering* consiste na segmentação de uma população heterogênea em subgrupos menos heterogêneos [Aldenderfer & Blashfield 1984]. Do ponto de vista *de baixo para cima* ou *bottom-up*, define-se *clustering* como a tarefa de encontrar grupos em um conjunto de dados considerando algum critério de similaridade [Duda & Hart 1973].

Fundamentalmente, *clustering* é considerada uma das principais tarefas relacionadas à mineração de dados (*data mining*) [Han & Kamber 2000], área de estudo que tem por objetivo a descoberta de padrões a partir de conjuntos de dados de tamanho considerável, buscando-se recuperar uma informação que possa ser compreendida pelo tomador de decisões [Chen et al. 1999].

Em um contexto mais prático, considerando-se a aplicação de métodos de *clustering*, ou mesmo técnicas de mineração de dados, pode-se objetivar a identificação de categorias ocultas (ou não-observadas) em relação a um conjunto de objetos reais ou mesmo artifici-

ais. Busca-se, a partir disto, a obtenção de informações relevantes ao tomador de decisão, seja este de qualquer área de atuação, como por exemplo a comercial ou a acadêmica [Blanchard et al. 2012, Blanchard et al. 2013].

Alguns exemplos de aplicação de métodos de *clustering* podem ser vistos em áreas como ciências naturais, engenharia, psicologia, medicina, marketing e economia. De forma a ilustrar tais aplicações, pode-se considerar os seguintes exemplos [Aloise & Hansen 2009]:

- Dentre o conjunto de dados mais popular na literatura para a avaliação de algoritmos de *clustering* está o Iris [Fisher 1936], no qual dados relativos a três espécies da flor Iris estão contidos. São estes: comprimento da sépala, largura da sépala, comprimento da pétala e largura da pétala. Utilizam-se, neste contexto, métodos de *clustering* para se determinar a espécie de plantas e animais a partir das amostras consideradas.
- Ao se desenvolver sistemas de recomendações de produtos em lojas virtuais, busca-se identificar aqueles consumidores com perfis similares, para que se possa estabelecer grupos entre eles. A partir disto, pode-se fazer predições acerca do interesse de dado indivíduo com base nas preferências de outros indivíduos de seu grupo, possibilitando-se que o sistema de venda online se reconfigure de modo automático e eficaz, mostrando com maior ênfase produtos que supostamente interessem a dado consumidor [Linden et al. 2003, Schafer et al. 2001].
- Ao formular métodos heurísticos aplicados ao roteamento de veículos [Ghiani et al. 2004], duas abordagens são comumente observadas: na abordagem *roteirize primeiro - agrupe depois*, ou *route-first – cluster-second*, uma rota que passa por todos os clientes é construída. Na sequência, esta rota é particionada em rotas viáveis para cada um dos veículos de modo a satisfazer as restrições de capacidade. Na estratégia *agrupe primeiro – roteirize depois*, ou *cluster-first – route-second*, diferentemente, particionam-se os clientes em grupos, para os quais a demanda total não supera a capacidade dos veículos a eles associados. Desta forma, resolve-se um problema do caixeiro viajante para cada um destes grupos [Applegate et al. 2007, Arenales et al. 2006]

Outros exemplos incluem pesquisas relacionadas à categorização de alimentos [Kohn et al. 2010, Ross & Murphey 1999], animais [Kelter et al. 1977], elementos léxicos [Miller 1969], bens duráveis [Urban et al. 1993], bens de consumo [Griffin & Hauser 1993], frases [Perkins 1993], dentre outros.

Ao se utilizar métodos de *clustering*, no entanto, assume-se geralmente que a percepção das categorias não-observadas é homogênea, tendo-se como entrada uma única matriz de dissimilaridades entre os objetos [Daws 1996]. Esta abordagem, por sua vez, pode não ser apropriada em certos casos, pois na realidade diferentes indivíduos podem julgar um mesmo conjunto de objetos com base em percepções diferenciadas, o que poderá resultar em arranjos distintos entre estes objetos [Blanchard et al. 2012].

Para demonstrar a inviabilidade de representar a heterogeneidade de forma apropriada em modelos de *clustering* mais tradicionais, considera-se como exemplo o Problema das p -Medianas (PPM). Este modelo é comumente utilizado neste contexto por considerar objetos reais como centros de *cluster* [Blanchard et al. 2012, Brusco & Kohn 2009, Brusco et al. 2012, Forgy 1965, Johnson 1965, Kohn et al. 2010, Mladenović et al. 2007, Siridov et al. 2008, Ward 1963].

Para este modelo, dado um conjunto de objetos, um subconjunto deste será selecionado como centros de segmento (exemplares ou medianas) e o restante dos objetos será alocado ao exemplar mais similar a cada um destes, de modo que algum critério seja otimizado, como por exemplo, a minimização da soma total das dissimilaridades entre os exemplares e os objetos a eles associados [Kohn et al. 2010]

Formalmente, dado um conjunto de $J(j, k = 1, 2, \dots, J)$ objetos, o problema das p -medianas pode ser definido com base em variáveis de decisão binárias e_{jk} , em que e_{jk} deverá ser 1 se um objeto j está associado a um objeto k , e 0 caso contrário. Os parâmetros para este problema incluem um valor para p , que representa o número de medianas a serem encontradas, e o custo d_{jk} relacionado à cada associação de par de objetos j e k . A partir da definição de tais elementos, o PPM é formulado como:

$$\text{Minimize } PM = \sum_{j=1}^J \sum_{k=1}^J d_{jk} e_{jk}, \quad (1.1)$$

sujeito a

$$\sum_{k=1}^J e_{jk} = 1, \quad \forall j = 1, \dots, J, \quad (1.2)$$

$$\sum_{j=1}^J e_{jj} = p, \quad (1.3)$$

$$e_{jk} \leq e_{kk}, \quad \forall j, k = 1, \dots, J, \quad (1.4)$$

$$e_{jk} \in \{0, 1\}, \quad \forall j, k = 1, \dots, J, \quad (1.5)$$

onde a função objetivo dada em (1.1) minimiza a dissimilaridade total entre todos os objetos e as respectivas medianas às quais estes foram associados. O conjunto de restrições dado por (1.2) implica que todo o objeto j deverá ser associado a exatamente um subconjunto, ou mediana. A restrição dada por (1.3) define que a solução do problema deverá ter p medianas (subconjuntos). Soluções com mais ou menos medianas do que este valor serão inviáveis. As restrições dadas pelas desigualdades (1.4) garantem que um objeto j somente poderá ser associado a um objeto k se este objeto k for uma mediana. Finalmente, as restrições dadas em (1.5) implicam que o valor das variáveis de decisão deverá ser, obrigatoriamente, binário.

Em sua essência, o modelo do PPM impõe em sua aplicação que se considere a percepção das categorias não-observadas como homogênea. Como mostrado no problema (1.1–1.5), os dados devem ser agregados, pois este modelo possui uma única matriz de proximidades (de similaridades ou dissimilaridades) como parâmetro [Blanchard et al. 2012, Blanchard et al. 2013, Daws 1996].

O problema é que os indivíduos normalmente diferem em relação à percepção de categorias em função de diversos fatores [Blanchard et al. 2012], como por exemplo, o conhecimento que determinado indivíduo detém acerca de um conjunto de objetos ou problema, sua idade, a finalidade para a qual o modelo está sendo empregado e o humor, entre outros [Medin & Schaffer 1978, Isen 2012, Suján & Dekleva 1987, John & Suján 1990, Ross & Murphey 1999].

De forma a ilustrar tal situação, considere o caso mostrado na Figura 1.1, na qual há três marcas bem conhecidas de chocolates - Twix, KitKat e Snickers. Para estas marcas, suponha que dois indivíduos, Indivíduo 1 e Indivíduo 2, tivessem de agrupá-las considerando algum critério de similaridade. O Indivíduo 1, por exemplo, colocou os chocolates Twix e KitKat na mesma categoria por considerar que ambos são crocantes. O chocolate Snickers, por sua vez, está em uma categoria diferente, por este indivíduo considerar que este chocolate não é crocante. O Indivíduo 2, diferentemente, colocou os chocolates Twix e Snickers em uma mesma categoria por considerar que ambos possuem recheio de caramelo. O chocolate KitKat foi atribuído a outra categoria, em razão de que este segundo indivíduo considera que este chocolate contém *wafers* em seu interior.

Ao considerar a informação contida na Figura 1.1 como entrada para o modelo do PPM, primeiramente uma matriz de dissimilaridades entre os objetos deverá ser calculada para cada um dos indivíduos. Então, agrega-se estas matrizes em apenas uma. No entanto, ao se realizar este processo, apenas uma estrutura de categorias poderá ser recu-

perada pelo PPM. Logo, não se poderá analisar a percepção de estruturas de categorias heterogêneas, as quais de fato existem para o exemplo considerado e são facilmente percebidas. Adicionalmente, não será possível identificar que existem grupos distintos de indivíduos, nem que estes grupos percebem as relações entre os objetos de forma distinta.

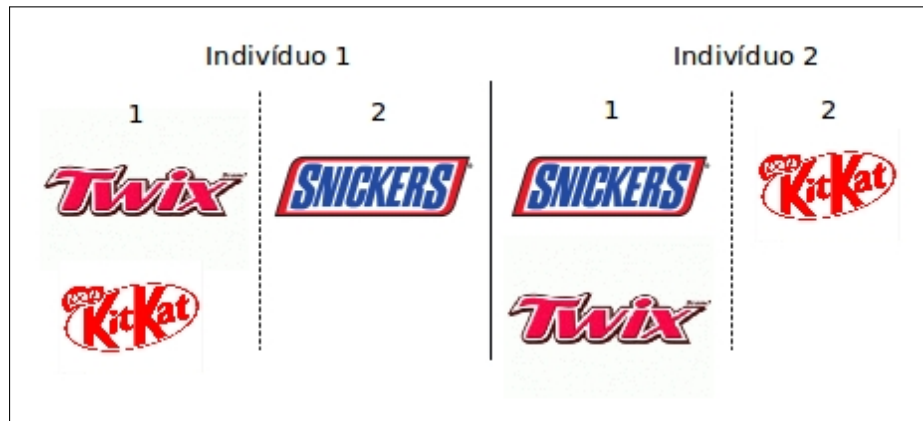


Figura 1.1: Exemplo de categorização heterogênea

Explorando a falta de modelos que considerem a heterogeneidade presente nos dados, Blanchard et al. (2012) propuseram um novo problema de otimização para análise por *cluster*. Seu modelo é baseado no clássico PPM, consistindo de um problema de programação não-linear inteira mista. O grande diferencial nesta nova abordagem é o uso de uma matriz de dissimilaridades para cada um dos indivíduos envolvidos no processo de análise dos dados, isto é, a agregação dos dados é dispensada.

A aquisição dos dados para seu modelo é baseada em uma "tarefa de triagem", cujo termo mais conhecido na literatura é o do inglês *sorting task*. A próxima seção deste texto explica em detalhes este processo de aquisição de dados e sua relação com o modelo de Blanchard et al. (2012), bem como detalhes acerca de sua formulação e suas limitações. Estas últimas fornecem alguns dos argumentos para justificar a proposta deste trabalho.

1.1 O problema das p -medianas heterogêneo

Blanchard et al. (2012) optaram por reformular o problema das p -medianas para que este tratasse de forma apropriada a heterogeneidade dos dados. Neste sentido, não há prova formal de que um modelo de *cluster* em particular seja o melhor [Kleinberg 2002], porém, para os propósitos considerados pelos autores, o problema das p -medianas mostrou-se apropriado em razão de (a) considerar objetos reais como centros de *cluster* e (b) ser um dos modelos de *clustering* mais utilizados e consolidados na literatura.

Segundo os autores, seu modelo de *clustering* difere dos clássicos em razão de possuir um propósito duplo. O primeiro destes é relativo à possibilidade de se identificar diversas categorias ocultas entre todos os objetos. Esta é uma grande diferença em relação ao problema clássico das p -medianas, que só permite identificar uma única estrutura de categorias. O segundo grande diferencial é que o modelo permite relacionar cada uma destas estruturas a um grupo distinto de indivíduos que compartilham uma opinião similar. Blanchard et al. (2012) sugerem que esta é uma representação mais realista, permitindo que esta relação seja explorada de diferentes maneiras quando da tomada de decisão.

Quanto à definição do problema, uma das primeiras considerações dos autores para a utilização de seu modelo é de que os dados de entrada são extraídos a partir de uma tarefa de triagem. Esta é uma técnica muito popular para se obter um grau de similaridade par-a-par entre objetos e permite entender a heterogeneidade na percepção dos indivíduos. A tarefa de triagem, ainda, é simples e seu funcionamento é de fácil entendimento [Courcoux et al. 2014], sendo vantajosa para os casos em que há um grande número de objetos, não sendo necessário o julgamento individual de cada um dos pares entre estes objetos. Relaciona-se, portanto, à uma atividade mais natural de percepção [Coxon 1999]. A tarefa de triagem também não impõe aos indivíduos um número fixo de categorias a serem feitas acerca dos objetos. Cada um destes indivíduos tem liberdade para criar sua própria estrutura [Viswanathan et al. 1999]. Também não há necessidade de que critérios sejam impostos para o processo de comparação.

Para ilustrar como esta técnica funciona, pode-se citar a aplicação apresentada por Blanchard et al. (2012), na qual 94 universitários foram instruídos a categorizar grandes redes de lojas dos Estados Unidos (21 no total), que atuam em diversos segmentos, de acordo com sua própria percepção de quão similar cada uma delas é entre si. Primeiramente, os estudantes deveriam revisar a lista das 21 alternativas, que foi apresentada em uma ordem aleatória diferente para cada um deles. Feito isso, solicitou-se que os estudantes classificassem as marcas em pilhas. Um número arbitrário de pilhas não foi imposto, assim como os critérios a serem utilizados.

Após a realização da categorização, uma pequena tarefa de distração foi dada aos estudantes. Estes deveriam contar quantas vezes a letra 'e' aparecia em um pequeno texto. Uma vez encerrada esta tarefa, realizou-se a segunda etapa do processo, na qual os indivíduos deveriam dizer quais foram as redes de lojas que estes acabaram de avaliar. Este processo serve para que, ao final da resolução do PPMH (Problema das p -medianas Heterogêneo), seja possível verificar se as medianas encontradas pelo modelo são de fato as marcas mais facilmente lembradas, isto é, lembradas em menos tempo. Obviamente a tendência é de que as marcas mais conhecidas sejam identificadas como medianas em

cada um dos segmentos.

De posse do resultado obtido por meio da tarefa de triagem, pôde-se então calcular uma matriz de similaridades para cada um dos indivíduos. Usou-se para este cálculo o método proposto por Takane (1980), no qual a similaridade entre dois objetos é inversamente proporcional à quantidade de objetos colocados na mesma pilha.

Formalmente, tem-se que W_{jl}^i é igual a 1 se o indivíduo i ($i = 1, 2, \dots, I$) coloca o objeto j ($j = 1, 2, \dots, J$) na pilha l ($l = 1, 2, \dots, c^i$), e 0 caso contrário. c^i corresponde à quantidade total de pilhas construídas pelo indivíduo i . Considerando-se que W^i é uma matriz $J \times c^i$ que representa a classificação feita pelo indivíduo i , então a matriz de similaridades S_{Wi} entre os J objetos para este indivíduo i é calculada como [Takane 1980]:

$$S_{Wi} = W^i((W^i)^T W^i)^{-1} (W^i)^T = [s_{jk}]_{J \times J}. \quad (1.6)$$

O valor proveniente deste cálculo varia dentro do intervalo $[0, 1]$. Pode-se, portanto, se obter o grau de dissimilaridade d_{jk}^i entre dois objetos j e k de acordo com a opinião do indivíduo i pelo complemento de s_{jk}^i para todo $j, k = 1, \dots, J$. Ou seja, uma matriz de dissimilaridades D_{Wi} para o indivíduo i é dada por

$$D_{Wi} = 1 - S_{Wi} = [d_{jk}]_{J \times J} \quad (1.7)$$

A partir da realização da tarefa de triagem, a formulação original do PPMH pode ser expressa em função dos seguintes parâmetros:

- d_{jk}^i é o grau de dissimilaridade observado entre os objetos j e k de acordo com a opinião do indivíduo i ;
- c^i é o número de pilhas feitas pelo indivíduo i na tarefa de triagem;
- δ é um fator de penalidade que necessita ser fixado pelo usuário do modelo de forma a condicionar o número de medianas a ser recuperado em cada segmento g ;
- G é o número total de segmentos a serem considerados na separação dos indivíduos em *clusters*.

As variáveis de decisão para este problema incluem:

- p^{ig} , que será 1 se o indivíduo i for designado ao segmento g ($g = 1, 2, \dots, G$), e 0 caso contrário ;
- e_{jk}^g , que será 1 se o objeto j for atribuído ao objeto k dentro do segmento g , e 0 contrário.;
- n^g , que é o número de medianas predito pelo modelo para cada segmento g .

A partir dos parâmetros e variáveis apresentados, formula-se o problema das p -medianas heterogêneo como:

$$\text{minimize } Z = \sum_{i=1}^I \sum_{g=1}^G p^{ig} \left[\sum_{j=1}^J \sum_{k=1}^J d_{jk}^i e_{jk}^g + \delta(c^i - n^g)^2 \right], \quad (1.8)$$

sujeito a

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (1.9)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (1.10)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (1.11)$$

$$\sum_{j=1}^J e_{jj}^g = n^g, \quad \forall g = 1, \dots, G, \quad (1.12)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (1.13)$$

$$p^{ig} \in \{0, 1\}, \quad \forall i = 1, \dots, I, \quad g = 1, \dots, G, \quad (1.14)$$

$$n^g \in \mathbb{Z}^+, \quad \forall g = 1, \dots, G, \quad (1.15)$$

onde o primeiro termo em (1.8) minimiza a soma das dissimilaridades entre os objetos e as medianas às quais estes objetos foram designados, considerando-se neste caso a pertinência dos indivíduos para cada segmento. O segundo termo da função objetivo minimiza a diferença entre o número de pilhas c^i que cada indivíduo fez na tarefa de triagem e o número de medianas predito para o segmento g ao qual o indivíduo i foi alocado.

O conjunto de restrições em (1.9) impõe que cada objeto j deverá estar associado a exatamente uma mediana em cada segmento g . O conjunto de restrições (1.10) impõe que cada um dos I indivíduos deverá estar atribuído a um segmento, exclusivamente. As restrições em (1.11) garantem que um objeto j só poderá ser atribuído a uma categoria dentro de um segmento cuja mediana é o objeto k se este for uma das medianas daquele segmento.

As restrições em (1.12) impõem que o valor da variável n^g , que representa o número total de medianas para cada segmento g , seja igual à contagem de objetos definidos como medianas em cada segmento g , isto é, os elementos na diagonal principal de cada uma das matrizes e^g . Finalmente, (1.13-1.14) são restrições de integralidade aplicadas ao modelo.

As restrições em (1.15), por sua vez, impõem que as variáveis n^g deverão possuir valores inteiros positivos.

Blanchard et al. (2012) diferenciam seu modelo em relação à versão clássica do problema das p-medianas em vários sentidos. Por exemplo, se o valor de n^g é desconhecido para todos os segmentos g , logo o problema está relacionado à seleção de um modelo. O problema proposto pelos autores considera não só este valor como desconhecido, mas também o valor de G . Uma alternativa para solucionar ambos os problemas, como sugerido pelo autores, seria a resolução das instâncias consideradas com valores diferentes para seus parâmetros, para que, então se analise o comportamento da função objetivo.

Sabidamente, este processo é oneroso, uma vez que muito tempo de computação pode ser necessário para que se resolva cada uma das instâncias. De forma a eliminar a necessidade do ajuste manual no número de medianas para cada segmento, isto é, o valor de n^g , os autores propuseram a inserção de um fator de penalidade δ na função objetivo, como descrito anteriormente.

Desta forma, o modelo deverá automaticamente selecionar o número de medianas n^g por segmento baseando-se no parâmetro c^i . Caso um indivíduo seja alocado a um segmento cujo número total de medianas difere em demasia da quantidade de pilhas feita por este indivíduo na tarefa de triagem, a função objetivo será penalizada. Esta opção por inserir um fator de penalidade, de acordo com os autores, é baseada na seguinte linha de raciocínio ([Blanchard et al. 2012]):

"...assumindo-se que o número de categorias, ou medianas, predito para um indivíduo, dada sua pertinência em um segmento, é x unidades acima ou abaixo do número de pilhas que esta pessoa criou na tarefa de triagem (c^i), Z será penalizada em δx^2

A penalidade inserida em Z , também de acordo com os autores, tem dois propósitos:

- Garantir que as estruturas de categoria identificadas serão similares em termos de complexidade cognitiva, buscando-se resgatar a estrutura observada empiricamente durante a tarefa de triagem, e;
- Simplificar o problema de seleção de um modelo, automatizando a escolha do número de categorias por segmento. Isto eliminará a necessidade de que se testem instâncias com diferentes valores usando um determinado número de medianas em cada segmento. Deste modo, apenas dois parâmetros necessitam de ajuste manual para a resolução do modelo, são eles: G e δ .

Blanchard et al. (2012) provaram a eficácia de seu modelo para o problema de representação da heterogeneidade presente nos dados. No entanto, observam-se algumas características neste modelo que o tornam um tanto complexo em sua aplicação, ou mesmo resolução. A primeira delas é a forma não-linear da função objetivo. Em seu trabalho, os autores resolveram o modelo (1.8-1.15) por meio de metaheurísticas, não sendo assim apresentados limitantes inferiores para que se analisasse o quão distante da solução ótima as soluções obtidas estavam.

O segundo ponto negativo neste modelo é que o ajuste do parâmetro δ (o fator de penalidade) exige grande experiência do usuário. Para este trabalho, apresenta-se um novo modelo para o PPMH, buscando-se minimizar o efeito destes aspectos negativos presentes no problema original. Desta forma, na próxima seção deste texto, os objetivos deste trabalho são apresentados, definindo-se o que nele será apresentado.

1.2 Objetivos

O principal objetivo deste trabalho é apresentar um novo modelo para o problema das p -medianas heterogêneo proposto por Blanchard et al. (2012), sendo que este novo modelo deverá ser não-paramétrico no que tange à definição do fator de penalidade δ . Este novo modelo compreenderá um modelo mais especializado e apropriado em termos práticos para os usuários finais (tomadores de decisão).

A fim de se atingir o objetivo principal descrito, consideram-se os seguintes objetivos específicos:

- *Objetivo específico 1:* obter formulações lineares inteiras mistas a partir do novo modelo proposto, de forma que seja possível obter limitantes inferiores para as instâncias a serem testadas neste trabalho;
- *Objetivo específico 2:* formular e implementar um algoritmo para resolver o modelo proposto para grandes instâncias de forma eficiente;
- *Objetivo específico 3:* validar o novo modelo e algoritmo propostos em termos de performance e recuperação da informação presente nos dados, comparando-se os resultados obtidos àqueles apresentados por Blanchard et al. (2012);
- *Objetivo específico 4:* apresentar uma aplicação do problema em questão, ilustrando assim seu potencial;
- *Objetivo específico 5:* analisar as contribuições e direções futuras desta pesquisa.

1.3 Estrutura e organização do texto

Este capítulo introduziu conceitos importantes ao entendimento e à justificativa deste trabalho. Destacou-se a importância do PPMH, proposto em [Blanchard et al. 2012], para tratar a presença da heterogeneidade em relação às diferentes estruturas de categorias observadas a partir da aplicação de uma tarefa de triagem. A contribuição dos autores é relevante, dados os diversos cenários para a tomada de decisão, nos quais sempre há divergência na forma como os indivíduos percebem o mundo.

Mostrou-se também que é relevante se obter um modelo simplificado para o problema apresentado pelos autores, buscando-se reduzir seu número de parâmetros. Blanchard et al. (2012) sugeriram a utilização de um parâmetro δ que define uma penalidade em seu modelo, a fim de que este possa recuperar o número de categorias presentes em cada um dos segmentos para os quais os indivíduos foram alocados. No entanto, definir este parâmetro exige certo esforço computacional, além de amplo conhecimento do modelo e problema.

Considerando-se a proposta deste trabalho, o restante deste texto está organizado como segue:

- O Capítulo 2 apresentará um novo modelo para o problema das p -medianas heterogêneo, o Problema das p -medianas Heterogêneo Livre de Penalidade (PPMHLP). Neste capítulo são também apresentadas duas novas formulações para o PPMHLP, ambas baseadas em técnicas de reformulação linear. Para estas duas novas formulações, resultados de experimentos computacionais são mostrados;
- O Capítulo 3 descreverá a metaheurística proposta neste trabalho para a resolução do PPMHLP de forma eficiente em cenários reais. Ao final do capítulo, resultados computacionais são dados, tanto para validar o novo modelo dado ao PPMH, o PPMHLP, bem como o algoritmo proposto para sua resolução. Um exemplo ilustrativo também é dado, no qual um caso real é mostrado;
- O Capítulo 4 apresenta resultados preliminares para uma nova versão do PPMHLP, estendendo-o a ambientes *fuzzy*;
- O Capítulo 5 traz uma análise global acerca deste trabalho e seus resultados, incluindo-se considerações acerca de sua continuidade.

Por fim, destaca-se que todos os algoritmos, códigos e instâncias descritos neste trabalho podem ser obtidos contactando-se o autor.

Capítulo 2

Reformulações para o problema das p -medianas heterogêneo livre de penalidade

Como descrito no capítulo 1, o modelo sugerido por Blanchard et al. (2012) para o problema das p -medianas heterogêneo apresenta algumas características que o tornam de difícil resolução e aplicação. Desta forma, neste capítulo será apresentado um novo modelo para este problema, o problema das p -medianas heterogêneo livre de penalidade - PPMHLP, no qual o fator de penalidade, principal parâmetro da formulação original, será eliminado.

Adicionalmente, duas novas formulações para o PPMHLP serão dadas. Ambas têm como objetivo tornar este novo modelo linear. A primeira delas será denominada PPMHLP1, consistindo de um modelo que utiliza uma quantidade de variáveis auxiliares relativamente pequena para se determinar o custo de associação entre os objetos j e suas respectivas medianas em cada segmento g , bem como calcular o número total de medianas por segmento g .

A segunda formulação será chamada de PPMHLP2, e incluirá um maior número de variáveis em relação à formulação PPMHLP1. Porém, esta permitirá a aplicação de algumas técnicas para a adição de restrições (cortes). Os experimentos computacionais realizados mostram que esta última opção influencia positivamente o desempenho de algoritmos de *branch-and-cut* na solução de algumas instâncias. A seção final deste capítulo apresenta uma análise acerca destes resultados.

2.1 O problema das p -medianas heterogêneo livre de penalidade

Sugere-se eliminar o parâmetro δ do PPMH impondo-se ao modelo que o número total de medianas para um dado segmento g seja sempre igual ao piso do número médio de pilhas feitas pelos indivíduos alocados a este segmento.

Pode-se modelar tal condição por meio do seguinte conjunto de restrições

$$\sum_{j=1}^J e_{jj}^g = \left\lfloor \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}} \right\rfloor, \quad \forall g = 1, \dots, G, \quad (2.1)$$

o qual, por sua vez, pode ser reescrito como

$$\sum_{j=1}^J e_{jj}^g \leq \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}}, \quad \forall g = 1, \dots, G, \quad (2.2)$$

uma vez que o processo de otimização tende a usar o maior número possível de medianas em cada segmento e que o lado esquerdo da desigualdade (2.2) sempre será um número inteiro. Desta forma, tem-se que no máximo este operador será igual ao piso do lado direito desta restrição.

Esta sugestão segue a linha de pensamento de diversos pesquisadores relacionados à literatura comportamental, os quais mostram que os indivíduos tendem a seguir regras simples quando analisam percepções e preferências acerca de diferentes marcas [Bettman & Park 1980, Bettman et al. 1998, Simon 1955, Shugan 1980], o que pode levar à obtenção de um número de categorias similar entre os indivíduos.

Pode-se também impor que o limite máximo para o número de medianas em cada segmento g seja menor ou igual ao valor inteiro mais próximo do número médio de pilhas feito pelos indivíduos alocados a este segmento. Para tal, soma-se 0,5 ao lado direito da desigualdade (2.2). Como o lado esquerdo desta desigualdade sempre será um número inteiro, esta condição também será garantida.

Por fim, sugere-se adicionar outra nova restrição ao modelo, dado por

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad (2.3)$$

A partir das manipulações apresentadas, pode-se reescrever o modelo do PPMH em uma forma não-paramétrica, exceto pelo parâmetro G . Ou seja, o PPMHLP é definido

como:

$$\text{Minimize } M = \sum_{i=1}^I \sum_{g=1}^G p^{ig} \left[\sum_{j=1}^J \sum_{k=1}^J d_{jk}^i e_{jk}^g \right] \quad (2.4)$$

sujeito a

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (2.5)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (2.6)$$

$$\sum_{j=1}^J e_{jj}^g \leq \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}}, \quad \forall g = 1, \dots, G, \quad (2.7)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.8)$$

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad (2.9)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.10)$$

$$p^{ig} \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, G, \quad (2.11)$$

em que a função objetivo dada por (2.4) minimiza a soma das dissimilaridades entre os objetos e as medianas às quais estes objetos foram designados, considerando-se neste caso a pertinência dos indivíduos para cada segmento, assim como no primeiro termo da expressão (1.8). As restrições em (2.5) impõem que cada objeto j deverá estar ligado a uma, e somente uma, categoria em cada um dos segmentos g . As restrições em (2.6) impõem que todo o indivíduo i deverá estar associado a um, e somente um, segmento g , obrigatoriamente.

As restrições em (2.7) garantem que o número total de medianas para cada segmento g seja menor ou igual ao piso da média do número de pilhas feitas pelos indivíduos alocados naquele segmento. Neste caso, o processo de otimização irá garantir que o número de medianas para cada segmento não será muito inferior ao valor médio do número de pilhas, uma vez que quanto mais medianas houver em cada segmento g , menor será o custo da função objetivo.

As restrições dadas por (2.8) garantem que um objeto j só poderá ser um exemplar da categoria k dentro do segmento g se o objeto k for uma mediana dentro daquele grupo. A desigualdade em (2.9) garantirá que nenhum *cluster* poderá estar vazio, isto é, sem ao

menos um indivíduo atribuído. Por fim, as restrições em (2.10) e (2.11) são restrições de integralidade.

O novo modelo apresentado em (2.4–2.11) elimina do modelo original a necessidade de se definirem valores para δ , assim como a necessidade de se analisar o custo da função objetivo obtido para diferentes valores de δ a fim de se ajustar o número de medianas recuperado pelo modelo. No entanto, o problema continua sendo não-linear, como se pode ver no modelo (2.4–2.11), no qual se tem o produto das variáveis de decisão p e e .

Para o modelo (2.4–2.11), foram feitos experimentos utilizando-se o pacote de otimização Couenne [Andreas et al. 2014], obtendo-se apenas soluções sub-ótimas para as quais o limitante inferior obtido foi nulo. Estas instâncias testadas são demasiadamente pequenas, com $I = 5$, $J = 5$ e $G = 2$. O computador utilizado nestes experimentos possui 62GB de memória e 12 processadores, sendo o tempo limite de execução fixado em 24 horas. O solver Couenne, no entanto, não se utiliza de paralelismo.

Dados os resultados observados a partir deste novo modelo, as próximas seções deste texto apresentarão algumas melhorias. Este modelo será linearizado, eliminando-se o produto das variáveis de decisão p e e , o que permitirá a obtenção de uma nova formulação, que por sua vez poderá levar à obtenção de soluções exatas via resolvedores comerciais de Programação Linear Inteira Mista (PLIM), como por exemplo o IBM CPLEX [IBM 2014]. O processo de reformulação a ser apresentado não eliminará a solução ótima do problema original descrito em (2.4–2.11).

O modelo também passará pela adição de novas restrições (cortes), a fim de acelerar sua resolução por meio de algoritmos de *branch-and-cut*, no intuito de se obterem limitantes inferiores para algumas instâncias. Estes limitantes servirão também para validar um algoritmo baseado na metaheurística VNS (*Variable Neighborhood Search*), também conhecida como busca em vizinhança variada, que será apresentado como alternativa de resolução do PPMHLP para grandes instâncias.

Observação 1 Se as variáveis e_{kk}^g são fixadas como binárias para todo g e k e a dissimilaridade entre uma mediana e ela mesma é nula, as variáveis e_{jk}^g , para todo $j \neq k$ podem ser relaxadas para o intervalo $[0, 1]$, uma vez que cada objeto j será atribuído à mediana mais próxima (menor dissimilaridade).

2.2 A formulação PPMHLP1

O custo de associar um objeto j a outro objeto k em um segmento g é dado pela soma das dissimilaridades atribuídas a este par de objetos por todos os indivíduos que

estão alocados neste segmento. Pode-se atribuir o custo de tal ligação a uma variável $c_{jk}^g \in [0, I - G + 1]$. Para tal, adiciona-se ao modelo (2.4–2.11) a restrição

$$\sum_{i=1}^I p^{ig} d_{jk}^i \leq c_{jk}^g + (1 - e_{jk}^g)(I - G + 1), \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.12)$$

em que, quando a variável e_{jk}^g assumir valor 1, isto é, o elemento j for associado ao objeto k dentro do segmento g , o segundo termo do lado direito desta desigualdade será anulado. Esta condição implica que, para que a solução do problema seja viável, c_{jk}^g deverá ser igual à soma das dissimilaridades relacionada a estes dois objetos, considerando os valores de d_{jk}^i para todos os indivíduos alocados ao segmento g . Caso contrário, quando $e_{jk}^g = 0$, o segundo termo desta desigualdade corresponderá a um valor suficientemente grande $I - G + 1$ (*Big M*), permitindo ao *solver* atribuir um valor zero a c_{jk}^g em razão do processo de minimização. Desta forma, o custo de ligar um objeto j a outro objeto k dentro do segmento g será nulo.

A partir da introdução desta nova variável ao modelo original do problema, podemos modificar a sua função objetivo dada em (2.4) para

$$\text{Minimize } M = \sum_{g=1}^G \sum_{j=1}^J \sum_{k=1}^J c_{jk}^g, \quad (2.13)$$

eliminando-se assim o produto das variáveis p e e . Nesta nova formulação, as variáveis e_{jk}^g , para todo $j \neq k$, não poderão ser relaxadas para o intervalo $[0, 1]$ quando as variáveis e_{jj}^g forem binárias. Caso isto seja feito, o custo da solução será sempre zero, uma vez que o segundo operando do lado direito da desigualdade (2.12) assumirá um valor suficientemente grande.

Observa-se a partir deste ponto que o PPMHLP continua não-linear, apesar da modificação da função objetivo, dado o conjunto de restrições definidas em (2.7). A fim de linearizar esta restrição, pode-se passar o denominador do lado direito para o lado esquerdo desta desigualdade, obtendo-se então uma nova restrição dada por

$$\sum_{i=1}^I \sum_{j=1}^J p^{ig} e_{jj}^g \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G. \quad (2.14)$$

Obtém-se, novamente, o produto das variáveis binárias p e e , para todo $j = k$. No entanto, o produto $e_{jj}^g * p^{ig}$ somente terá valor igual a 1 se ambas variáveis tiverem valor 1. Caso contrário, o valor deste produto será zero. Pode-se aplicar neste caso a técnica de

reformulação dada em [Fortet 1960], na qual substitui-se $e_{jj}^g * p^{ig}$ por w_j^{ig} ($w_j^{ig} \in [0, 1]$), devendo-se adicionar ao modelo três novas restrições que garantam que $\max\{0, p^{ig} + e_{jj}^g - 1\} \leq w_j^{ig}$. São elas:

$$w_j^{ig} \leq e_{jj}^g, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (2.15)$$

$$w_j^{ig} \leq p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (2.16)$$

$$w_j^{ig} \geq p^{ig} + e_{jj}^g - 1, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J. \quad (2.17)$$

Desta forma, as restrições em (2.14) podem ser reescritas como

$$\sum_{i=1}^I \sum_{j=1}^J w_j^{ig} \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G. \quad (2.18)$$

A formulação PPMHLP1 completa é apresentada a seguir:

$$\text{Minimize } M = \sum_{g=1}^G \sum_{j=1}^J \sum_{k=1}^J c_{jk}^g, \quad (2.19)$$

sujeito a

$$\sum_{i=1}^I p^{ig} d_{jk}^i \leq c_{jk}^g + (1 - e_{jk}^g)(I - G + 1), \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.20)$$

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (2.21)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (2.22)$$

$$\sum_{i=1}^I \sum_{j=1}^J w_j^{ig} \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G, \quad (2.23)$$

$$w_j^{ig} \leq e_{jj}^g, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (2.24)$$

$$w_j^{ig} \leq p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (2.25)$$

$$w_j^{ig} \geq e_{jj}^g + p^{ig} - 1, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (2.26)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.27)$$

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad (2.28)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.29)$$

$$p^{ig} \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad (2.30)$$

$$c_{jk}^g \in [0, I - G + 1], \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.31)$$

$$w_j^{ig} \in [0, 1], \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J. \quad (2.32)$$

A função objetivo dada em (2.19) minimiza o grau de dissimilaridade entre todos os objetos j e k , para todo segmento g , condicional à pertinência dos indivíduos i para cada um destes segmentos. O conjunto de restrições dadas em (2.20) fará com que a variável que denota o custo de ligação entre dois objetos j e k dentro de um segmento g , considerando-se a pertinência dos indivíduos naquele segmento, seja igual à soma de todas as dissimilaridades atribuídas a este par de objetos pelos indivíduos alocados a este segmento. A restrição dada por (2.21) garante que cada um dos objetos j deverá estar ligado a exatamente um objeto k , isto para todos os segmentos g . Neste caso, o índice k refere-se sempre aos objetos atribuídos como medianas para cada segmento.

As restrições dadas em (2.22) impõem que cada indivíduo i seja alocado a exatamente um segmento g . O conjunto de restrições em (2.23) garante que o número máximo de medianas para cada segmento g seja no máximo igual ao piso do número médio de pilhas feitas pelos indivíduos i alocados a este segmento. Deve-se notar que a partir da aplicação do processo de reformulação descrito, este conjunto de restrições passou a ser linear. O conjunto de restrições dadas por (2.24–2.26) garante que w_j^{ig} será 1 quando ambas as variáveis e_{jj}^g e p^{ig} forem iguais a 1, para todo $g = 1, \dots, G$, $i = 1, \dots, I$ e $j = 1, \dots, J$, e 0 caso contrário. Obtém-se, desta forma, uma versão linear inteira do PPMHLP.

As restrições dadas por (2.27) impõem que um objeto j só poderá estar associado a outro objeto k em um segmento g se o objeto k for uma mediana dentro deste segmento. O conjunto de restrições em (2.28) garantirá que nenhum segmento g fique vazio, isto é, sem ao menos um indivíduo i alocado. As restrições em (2.29) e (2.30) são restrições de integralidade para as variáveis e e p .

Em (2.31), limita-se o valor mínimo e valor máximo que as variáveis c_{jk}^g , que representam o custo da ligação entre um objeto j e outro objeto k , podem assumir dentro de um segmento g . Utiliza-se como limite superior o valor de $I - G + 1$, uma vez que no máximo a dissimilaridade atribuída por um indivíduo i a um par de objetos é 1 e que ao menos um indivíduo i deverá ser alocado a cada segmento g , dado à restrição em (2.28). Logo, a soma de todas as dissimilaridades do par de objetos (j, k) , considerando-se todos

os indivíduos i , será no máximo igual a $I - G + 1$. Por fim, (2.32) restringe todas as variáveis w_j^{ig} no intervalo $[0, 1]$, uma vez que seus valores sempre serão ou 0 ou 1, dadas as restrições em (2.24–2.26).

Esta nova formulação para o PPMHLP foi testada computacionalmente, sendo seus resultados dispostos ao final deste capítulo. Comparam-se tais resultados com os resultados obtidos a partir da formulação PPMHLP2. Esta última é apresentada na próxima Seção deste texto.

Observação 2 *Sabe-se a priori que todas as variáveis de decisão e_{jk}^g deverão possuir valor integral na solução ótima do problema (2.19–2.32), para todo $g = 1, \dots, G$ e $j, k = 1, \dots, J$, a fim de satisfazer às restrições dadas em (2.29). No entanto, o processo de otimização sempre buscará colocar valores fracionários nestas variáveis, uma vez que estes implicam em custo menor, ou mesmo nulo, para cada associação de objetos j e k em todos os segmentos g . Isto é, as restrições dadas em (2.20) diminuirão consideravelmente o desempenho dos algoritmos de branch-and-bound para este problema, dado o número total de nós de ramificação que serão necessários para o algoritmo gerar uma solução que satisfaça a (2.29) para todas estas variáveis.*

2.3 A formulação PPMHLP2

A formulação PPMHLP1 mostra-se como alternativa ao problema (2.4–2.11). Embora esta versão seja linear, o problema descrito na Observação 2 influencia negativamente a resolução do modelo via *solvers* que utilizam algoritmos *branch-and-bound*. Para minimizar este problema, sugere-se novamente aplicar a técnica de reformulação descrita em [Fortet 1960] ao modelo (2.4–2.11), estendendo-a a todos os produtos $e_{jk}^g * p^{ig}$.

Como mostrado anteriormente, o produto $e_{jk}^g * p^{ig}$ somente será igual a 1 se ambos os termos e_{jk}^g e p^{ig} forem iguais a 1. Caso contrário, o valor do produto será zero. Logo, pode-se substituir estes produtos por um conjunto de variáveis $w_{jk}^{ig} \in [0, 1]$, para todo $g = 1, \dots, G$, $i = 1, \dots, I$, $j = 1, \dots, J$ e $k = 1, \dots, J$, juntamente com as restrições que garantam que $\max\{0, e_{jk}^g + p^{ig} - 1\} \leq w_{jk}^{ig}$. São elas:

$$w_{jk}^{ig} \leq e_{jk}^g, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (2.33)$$

$$w_{jk}^{ig} \leq p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (2.34)$$

$$w_{jk}^{ig} \geq e_{jk}^g + p^{ig} - 1, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, G, \quad j, k = 1, \dots, J. \quad (2.35)$$

Dado que e_{jk}^g e p^{ig} são variáveis binárias, as desigualdades introduzidas em (2.33–2.35) garantem que w_{jk}^{ig} só poderá ser igual a 1 quando ambas as variáveis tiverem valor 1, e 0 caso contrário. A partir da inclusão destas mudanças, a função objetivo dada por (2.4) pode ser reescrita como

$$\text{Minimize } M = \sum_{g=1}^G \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^J d_{jk}^i w_{jk}^{ig}, \quad (2.36)$$

assim como o conjunto de restrições (2.7) pode ser novamente reescrito na forma

$$\sum_{i=1}^I \sum_{j=1}^J w_{jj}^{ig} \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G. \quad (2.37)$$

Esta alteração no modelo (2.4–2.11) permite ainda acelerar o processo de otimização por meio da adição de novas restrições ao modelo. Tais cortes, no entanto, não afetarão a solução ótima do problema original. Baseando-se na técnica de reformulação e linearização descrita em [Sherali & Alameddine 1992], pode-se obter um conjunto adicional de restrições por meio da multiplicação das $J \times G$ restrições dadas em (2.5) por p^{ig} , para todo i e g . Portanto, substituindo-se os produtos $e_{jk}^g * p^{ig}$ por w_{jk}^{ig} (equivalentes), para todo i, j, k e g , o novo conjunto de restrições será escrito como

$$\sum_{k=1}^J w_{jk}^{ig} = p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J. \quad (2.38)$$

A adição destas novas restrições à formulação cria redundâncias, permitindo-se sua simplificação. Pode-se eliminar do modelo as restrições dadas em (2.34), pois a soma de todos os termos do lado esquerdo da igualdade em (2.38) só poderá ser igual a p^{ig} se todos os termos w_{jk}^{ig} forem menores ou iguais a p^{ig} , para todo i, g, j e k .

Pode-se também eliminar do modelo as restrições dadas em (2.35), pois como as variáveis e são binárias e o PPMHLP contém a restrição

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (2.39)$$

apenas uma dos termos w_{jk}^{ig} no lado esquerdo da igualdade (2.38) poderá assumir 1, e isto somente poderá ocorrer quando e_{jk}^g for 1. Adicionalmente, como a restrição em (2.38) força que a soma dos termos de seu lado esquerdo seja igual a p^{ig} , um dos termos w_{jk}^{ig} obrigatoriamente deverá ser 1, e somente o poderá ser quando p^{ig} for igual a 1. Logo,

vê-se que w_{jk}^{ig} assumirá valor 1 sempre que ambos e_{jk}^g e p^{ig} forem 1, e 0 caso contrário. Esta condição equivale exatamente ao que a restrição dada em (2.35) impõe.

A partir das manipulações apresentadas, a nova formulação PPMHLP2 é mostrada integralmente a seguir:

$$\text{Minimize } S = \sum_{g=1}^G \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^J d_{jk}^i w_{jk}^{ig}, \quad (2.40)$$

sujeito a:

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (2.41)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (2.42)$$

$$w_{jk}^{ig} \leq e_{jk}^g, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (2.43)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (2.44)$$

$$\sum_{i=1}^I \sum_{j=1}^J w_{jj}^{ig} \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G, \quad (2.45)$$

$$\sum_{k=1}^J w_{jk}^{ig} = p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (2.46)$$

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad (2.47)$$

$$w_{jk}^{ig} \in [0, 1], \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (2.48)$$

$$e_{jj}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (2.49)$$

$$e_{jk}^g \in [0, 1], \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad j \neq k, \quad (2.50)$$

$$p^{ig} \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I. \quad (2.51)$$

Ambas as formulações, PPMHLP1 e PPMHLP2, podem ser executadas via resolvidores, como por exemplo o IBM CPLEX [IBM 2014]. Os resultados para experimentos com estes modelos e resolvidor são apresentados na próxima seção deste texto.

Pode-se avaliar, a partir destes resultados, qual das duas formulações apresentadas fornece melhores limitantes inferiores para o PPMHLP, julgando-se assim a qualidade de soluções obtidas heurísticamente ou pelo próprio resolvidor. Tal possibilidade não

foi apresentada em [Blanchard et al. 2012] para o modelo (1.8–1.15). Não se observou também tal possibilidade a partir de testes computacionais com o modelo (2.4–2.11).

2.4 Resultados computacionais

Blanchard et al. (2012) utilizaram uma Simulação de Monte Carlo para validar seu modelo, gerando um conjunto de 27 instâncias. Para estas, os valores das variáveis e e p a serem recuperados pelo modelo são previamente conhecidos dado o processo de simulação. Desta forma, pode-se avaliar o quão próximo deste resultado esperado a resposta do modelo está. Como as instâncias geradas neste processo possuem diferentes arranjos em relação às suas estruturas de categorias, número total de segmentos e número total de indivíduos, pode-se também avaliar o grau de sensibilidade do modelo e seus algoritmos a estes fatores.

Adicionalmente, após o cálculo das matrizes de dissimilaridades D_{wi} para todos os indivíduos i , os autores adicionaram pequenos ruídos a tais matrizes. Distorções também foram adicionadas ao número total de pilhas c^i atribuído a cada um dos indivíduos i . Para ambos os casos, as perturbações seguem distribuições normais $N(\bar{M}, DP)$, com média (\bar{M}) e desvio padrão (DP) variados. Estas distorções também são consideradas como fatores de influência na análise de sensibilidade do PPMHLP e seus algoritmos.

A Tabela 2.1 apresenta a estrutura de cada uma das instâncias simuladas geradas por Blanchard et al. (2012). Optou-se por seguir a mesma metodologia dos autores para a realização de testes com o novo modelo, podendo-se desta forma comparar o PPMHLP ao PPMH quantitativamente, observando-se eventuais diferenças em termos de qualidade e desempenho. As 27 instâncias aqui utilizadas são as mesmas utilizadas pelos autores.

O significado dos valores apresentados em cada uma das colunas da Tabela 2.1 é explicado nos itens que seguem:

- Coluna *Instância*: código de identificação de cada uma das instâncias. Este código será utilizado deste ponto em diante, uma vez que tais instâncias, somadas a um conjunto de dados reais, são os dados de entrada para todo o restante deste trabalho;
- Coluna *Indivíduos*: quantidade total de indivíduos I da referida instância;
- Coluna *Segmentos*: número total de segmentos que o modelo deverá recuperar, isto é, grupos aos quais os I indivíduos deverão ser alocados;
- Coluna *Objetos*: quantidade total J de objetos a serem categorizados, isto é, definidos como medianas e seus satélites, para cada um dos G segmentos;
- Coluna *Categorias*: número de medianas definidos na simulação para cada um dos

Instância	Indivíduos <i>I</i>	Segmentos <i>G</i>	Objetos <i>J</i>	Categorias	Distorções Dissimilaridades	Distorções Pilhas
1	150	10	30	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)
2	300	2	18	Todas 6	N(0, 0.1)	0
3	450	2	18	50 % 3, 50 % 6	N(0, 0.05)	0
4	150	2	18	Todas 3	N(0, 0.05)	N(0, 0.5)
5	450	10	18	Todas 6	N(0, 0.05)	N(0, 1)
6	150	10	18	50 % 3, 50 % 6	N(0, 0.05)	0
7	300	2	18	Todas 6	0	N(0, 0.5)
8	150	10	18	50 % 3, 50 % 6	0	N(0, 1)
9	300	10	30	Todas 3	N(0, 0.05)	N(0, 0.5)
10	450	6	18	Todas 3	N(0, 0.1)	N(0, 1)
11	150	6	30	Todas 6	N(0, 0.1)	0
12	300	10	18	Todas 3	0	0
13	450	10	18	Todas 6	N(0, 0.1)	0
14	300	6	18	50 % 3, 50 % 6	0	N(0, 1)
15	300	2	30	Todas 6	N(0, 0.05)	N(0, 1)
16	450	2	30	50 % 3, 50 % 6	0	N(0, 1)
17	300	6	18	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)
18	300	6	30	50 % 3, 50 % 6	N(0, 0.05)	0
19	150	6	18	Todas 6	0	N(0, 0.5)
20	450	6	30	Todas 3	0	0
21	150	2	30	Todas 3	N(0, 0.1)	N(0, 1)
22	450	2	18	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)
23	450	6	18	Todas 3	N(0, 0.05)	N(0, 0.5)
24	300	10	18	Todas 3	N(0, 0.1)	N(0, 1)
25	150	6	18	Todas 6	N(0, 0.05)	N(0, 1)
26	150	2	18	Todas 3	0	0
27	450	10	30	Todas 6	0	N(0, 0.5)

Tabela 2.1: Estrutura das instâncias simuladas

segmentos g . Geraram-se instâncias com 3 e 6 estruturas de categorias. Híbridos entre estes dois valores também foram gerados, a fim de se obterem instâncias de resolução dificultada;

- Coluna *Distorções/Dissimilaridades*: nível de ruído adicionado às matrizes de dissimilaridades D_{wi} para cada um dos indivíduos i . Cada uma destas matrizes recebeu um conjunto de valores diferente para cada um dos indivíduos considerando-se a distribuição normal mostrada na tabela;
- Coluna *Distorções/Pilhas*: nível de ruído adicionado aos valores gerados para c^i para cada um dos indivíduos i , isto é, valores mostrados na Coluna *Categorias* da Tabela 2.1. Cada uma das variáveis c^i recebeu um valor de ruído diferente.

De posse das 27 instâncias definidas, as formulações PPMHLP1 e PPMHLP2 foram executadas no *solver* CPLEX [IBM 2014], versão 12.5. Para cada instância, definiu-se 24 horas como tempo limite. O computador utilizado possui 12 processadores Intel(R)

Xeon(R) CPU X5650 de 2.67GHz e 62GB de memória RAM. Utilizou-se também a configuração padrão do *solver*, a qual permite que o mesmo utilize 12 *threads* paralelas para cada uma das instâncias (uma *thread* por processador).

Destaca-se que apenas uma instância foi resolvida por vez, permitindo-se ao *solver* utilizar todos os recursos computacionais disponíveis para cada uma destas. A Tabela 2.2 apresenta os resultados obtidos a partir do experimento. Cada uma das colunas desta tabela é explicada nos itens que seguem:

- A Coluna *Instância*: esta coluna refere-se ao código das instâncias geradas por simulação, descritas anteriormente nesta Seção;
- A Coluna *PPMHLP2/Melhor Custo Solução Inteira*: esta coluna apresenta o custo da melhor solução inteira obtida para o problema a partir da resolução de cada instância pela formulação PPMHLP2 após 24 horas de execução;
- A Coluna *PPMHLP2/Menor Custo Relaxação*: esta coluna apresenta o menor custo obtido pela resolução da relaxação do problema, considerando a resolução de todas relaxações em todos os nós da árvore do algoritmo de *branch-and-bound* dentro do limite de 24 horas de processamento;
- A Coluna *PPMHLP2/GAP*: esta coluna mostra a distância, em porcentagem, entre o menor custo de relaxação obtido dentre todos os nós resolvidos pelo algoritmo de *branch-and-bound* e a melhor solução inteira obtida;
- A Coluna *PPMHLP2/Nós Explorados e PPMHLP1/Nós Explorados* : número de nós explorados pelo *solver* durante a resolução;
- A Coluna *PPMHLP1/Melhor Custo Solução Inteira*: esta coluna apresenta o custo da melhor solução inteira obtida para o problema a partir da resolução de cada instância pela formulação PPMHLP1 após 24 horas de execução;
- A Coluna *PPMHLP1/GAP*: esta coluna mostra a distância, em porcentagem, entre o menor custo de relaxação obtido a partir da formulação PPMHLP2 e a melhor solução inteira obtida por meio da formulação PPMHLP1.

Instância	PPMHLP2				PPMHLP1		
	Melhor Custo Solução Inteira	Menor Custo Relaxação	GAP	Nós Explorados	Melhor Custo Solução Inteira	GAP	Nós Explorados
1	-	-	100.00%	-	3507.7327	100.00%	158485
2	2389.32036	2336.58209	2.21%	8310	2391.94615	2.31%	410248
3	4607.0683	4398.937717	4.52%	1097	5104.79194	13.83%	330278
4	1870.07439	1823.609552	2.48%	41186	1871.15249	2.54%	2966838
5	7353.25648	0	100.00%	1	4900.87911	100.00%	22200
6	2396.44816	1382.715212	42.30%	276	1762.59574	21.55%	82193
7	2651.01098	2413.333147	8.97%	3432	2651.01098	8.97%	1756295
8	1824.16658	1470.83555	19.37%	8	1585.16597	7.21%	84443
9	8425.27766	0	100.00%	1	7784.44564	100.00%	7443
10	7267.39799	4995.302978	31.26%	3	5889.9318	15.19%	58635
11	4235.06645	0	100.00%	1	2958.48186	100.00%	585748
12	4849.995	3749.985	22.68%	1	3779.9856	0.79%	374311
13	7330.23751	0	100.00%	1	4678.67604	100.00%	21621
14	4648.16355	2905.83791	37.48%	15	3227.33306	9.96%	354894
15	6087.37635	5644.11692	7.28%	677	5775.7719	2.28%	105515
16	10752.5	9670.02	10.07%	29	10204.2	5.23%	53782
17	4127.72272	2671.693225	35.27%	1601	3386.6488	21.11%	90194
18	8448.21217	0	100.00%	1	7073.90629	100.00%	14741
19	1269.00531	1190.00595	6.23%	342	1264.67202	5.90%	138893
20	12645	0	100.00%	1	11435.7	100.00%	16408
21	3726.85898	3405.44083	8.62%	111	3639.28456	6.43%	8291974
22	4710.93306	4233.501076	10.13%	1871	4906.24547	13.71%	188348
23	7262.72169	5312.943074	26.85%	5	5784.20158	8.15%	46099
24	4834.31262	3212.933781	33.54%	3	3983.76823	19.35%	83792
25	1460.81456	1141.321417	21.87%	1752	1363.56496	16.30%	147203
26	1874.9925	1874.9925	0.00%	1	1874.9925	0.00%	3330090
27	12690	0	100.00%	1	10103.4	100.00%	23794

Tabela 2.2: Resultados computacionais para as formulações PPMHLP1 e PPMHLP2 obtidos por meio do *solver* CPLEX 12.5

Calculou-se o GAP para a formulação PPMHLP1 com base nos valores de relaxação obtidos para a formulação PPMHLP2, pois a resolução da formulação PPMHLP1 não retornou quaisquer limitantes inferiores, além do trivial, mesmo após 24h de execução.

Os resultados dos experimentos, apresentados na Tabela 2.2, indicam que, apesar de o modelo do PPMHLP ter sido linearizado, este continuou de difícil resolução. Grande parte desta dificuldade pode ser atribuída ao fato de ambas as formulações apresentarem considerável número de variáveis de decisão. A Tabela 2.3 apresenta o número total de variáveis de decisão que as compõem, mostrando o caso genérico e um exemplo numérico, considerando-se a Instância 1 utilizada nos testes.

Variáveis (tipo)	PPMHLP1 Genérico	PPMHLP2 Genérico	PPMHLP1 Instância 1	PPMHLP2 Instância 1
e (binárias)	$G * J^2$	$G * J$	9000	300
e (reais)	-	$G(J^2 - J)$	0	8700
p (binárias)	$I * G$	$I * G$	4500	4500
w (reais)	$I * G * J$	$I * G * J^2$	135000	4050000
c (reais)	$G * J^2$	-	9000	0
Total Binárias	$G(J^2 + I)$	$G(J + I)$	13500	4800
Total Reais	$G(I * J + J^2)$	$G(J^2 - J + I * J^2)$	144000	4058700
Total	$G(2 * J^2 + I + I * J)$	$G(J + I + J^2 - J + I * J^2)$	157500	4063500

Tabela 2.3: Total de variáveis de decisão para PPMHLP1 e PPMHLP2

Soma-se a isto o fato de o problema das p -medianas, presente no modelo, ser NP-árduo [Kariv & Hakimi 1979]. Embora este problema atualmente seja resolvido de forma eficiente e ótima para um considerável número de variáveis, a complexidade e tamanho do PPMHLP como um todo impede a obtenção de soluções ótimas para instâncias de tamanho elevado, como as utilizadas neste trabalho.

Ao se analisar a Tabela 2.2, em relação aos resultados obtidos por meio da formulação PPMHLP1, pode-se verificar que esta apresenta melhores resultados quando o número de segmentos é alto, por exemplo $G = 10$. Nestes casos, a formulação PPMHLP2 ou não apresentou sequer uma solução (Instância 1) ou apresentou soluções de custo elevado.

Na verdade, estas soluções de custo elevado para $G = 10$ da formulação PPMHLP2 sequer fazem sentido ao se analisar a estrutura de categorias e segmentos de indivíduos por ela recuperados, pois nestas soluções para cada segmento g apenas um objeto j foi definido como mediana e todos os demais objetos foram associados a ele. Isto significa que não houve qualquer otimização, ou otimização satisfatória, da função objetivo em relação à solução inteira do problema.

Esta "vantagem" observada para a formulação PPMHLP1 deveu-se ao fato de que o *solver* conseguiu explorar diversos nós da árvore de resolução. Também, o *solver* con-

seguir melhores soluções inteiras por meio de suas heurísticas. O reduzido número de variáveis de decisão desta formulação (em relação ao PPMHLP2) permitiu a obtenção destes resultados.

Contrariamente, a formulação PPMHLP2 fez com que o *solver* não explorasse nenhum nó, além do nó raiz, da árvore de resolução em grande parte dos casos. Isto se deve ao fato de que esta formulação possui uma quantidade de variáveis de decisão significativamente maior que a formulação PPMHLP1. No entanto, em vários casos esta formulação retornou soluções inteiras de melhor custo, especialmente para as instâncias com menor número de segmentos ($G = 2$), onde o número de variáveis é um pouco menor.

A formulação PPMHLP2, evidentemente, mostrou-se mais forte que a formulação PPMHLP1, pois além de permitir a obtenção de soluções inteiras melhores que a PPMHLP1 em alguns casos, possibilitou a obtenção de limitantes inferiores para grande parte das instâncias. Estes limitantes são úteis, uma vez que permitirão uma avaliação de desempenho da metaheurística descrita no Capítulo 3 deste texto, a qual é uma alternativa eficiente de solução para o PPMHLP. Os valores apresentados na Tabela 2.2 serão considerados para a análise desta metaheurística.

Por fim, uma importante observação acerca dos resultados, é que para a Instância 26 obteve-se a solução ótima. Esta instância é a menos complexa dentre as 27, pois tem menor tamanho e não apresenta distorções nem no número de pilhas nem nas matrizes de dissimilaridades. Apesar disto, a otimalidade da solução mostrada em [Blanchard et al. 2012] não foi provada para esta instância para o modelo original, o PPMH.

Capítulo 3

Um algoritmo VNS para o problema das p -medianas heterogêneo livre de penalidade

Um problema de otimização combinatória pode ser descrito genericamente como [Mladenović et al. 2007]:

$$\min\{f(x)|x \in X\}, \quad (3.1)$$

em que $f(x)$ é a função objetivo a ser minimizada, ou maximizada, e X é um conjunto de soluções viáveis para este problema. Por sua vez, uma solução $x^* \in X$ é dita ótima se, e somente se,

$$f(x^*) \leq f(x), \forall x \in X. \quad (3.2)$$

Um algoritmo exato para o problema descrito em (3.1), caso exista, busca uma solução x^* , ao mesmo tempo em que busca a prova de sua otimalidade. Em certos casos, o algoritmo poderá mostrar que o problema ou não tem solução alguma ($X = \emptyset$) ou então que o problema tem infinitas soluções.

Dentre os mais conhecidos algoritmos exatos estão o método *simplex*, utilizado para problemas programação linear [Dantzig 1963], e o método *branch-and-bound* [Land & Doig 1960], utilizado na resolução de problemas de programação inteira. Este segundo, utiliza-se do método *simplex* em seus subproblemas. Em termos práticos, algoritmos exatos tradicionais nem sempre são eficientes para resolver certos problemas de otimização combinatória, e o PPMHLP é um exemplo disto. A eficiência destes algoritmos dependerá da complexidade do problema tratado ou mesmo do tamanho das instâncias consideradas,

como demonstrado no Capítulo 2.

Mostrou-se, ainda no Capítulo 2, que é impraticável resolver ambas as formulações obtidas para o PPMHLP, a PPMHLP1 e a PPMHLP2, por meio dos referidos métodos exatos. Isto, mesmo para instâncias geradas por simulação, que em teoria seriam resolvidas mais facilmente devido ao elevado grau de separação entre os objetos e indivíduos. Portanto, vê-se que é necessário recorrer a métodos de outra natureza para que se possa tratar o problema em cenários reais. Para tal, a literatura sugere que heurísticas e metaheurísticas são abordagens adequadas.

Neste capítulo, apresenta-se uma breve descrição de tais abordagens, bem como formula-se um algoritmo baseado na metaheurística VNS, especificamente direcionada ao PPMHLP, o VNS-PPMHLP (Algoritmo de Busca em Vizinhança Variada para o Problema das p -Medianas Heterogêneo Livre de Penalidade). Resultados computacionais também são mostrados ao final deste capítulo, bem como a validação destes via análise dos dados da Simulação de Monte Carlo. Uma aplicação do modelo a cenários reais também é apresentada.

3.1 Heurísticas e metaheurísticas

Um algoritmo heurístico, de acordo com Mladenović et al. (2007), é um algoritmo que resolve um problema de otimização, como por exemplo (3.1), de maneira a encontrar uma solução x' que seja muito próxima à solução ótima x^* . Tal proximidade é avaliada em função do custo obtido para a função objetivo em ambos os casos, ou seja, avalia-se a relação entre $f(x')$ e $f(x^*)$. O valor de $f(x^*)$, em termos práticos, é geralmente desconhecido.

Metaheurísticas compreendem uma família de métodos complementares às heurísticas, sendo utilizadas não só para encontrar uma solução x' de boa qualidade para dado problema, mas sim seu ótimo global x^* . Tal possibilidade existe, pois as metaheurísticas se utilizam de diferentes estratégias de busca aleatória. Estas estratégias de busca aleatória são necessárias para evitar que algoritmos heurísticos fiquem presos em ótimos locais (x''), soluções que podem não ser o ótimo global do problema. Estes, além de serem soluções viáveis, são ótimos globais em relação às suas respectivas vizinhanças. Uma vizinhança, por sua vez, consiste no conjunto de todas as soluções obtidas a partir de uma solução x' qualquer, por meio de alguma estratégia de movimentação [Lee & Geem 2005, Jones et al. 2002].

Por exemplo, pode-se considerar como vizinhança de uma solução x' , assumindo-se que esta deva ser composta apenas de valores binários, todas as soluções obtidas por meio

da complementação de cada um de seus componentes. Destaca-se, ainda, que um ótimo local pode ser também o ótimo global do problema em questão.

Uma importante consideração a ser feita é que, uma vez que um algoritmo heurístico tradicional encontre um ótimo local, tal algoritmo é incapaz de melhorar a solução corrente a partir deste ponto. No entanto, ótimos globais podem ser encontrados por meio de metaheurísticas, pois estes métodos conseguem escapar destes ótimos locais. O problema, neste caso, é que estes métodos não fornecem prova de otimalidade para as soluções obtidas.

3.2 O algoritmo de busca vizinhança variada

Um método de busca local para um problema de otimização combinatória executa uma sequência de mudanças a partir de uma solução inicial dada a este problema. Nestes métodos, busca-se melhorar o valor da função objetivo destas soluções a cada passo, até que um ótimo local seja encontrado [Mladenović & Hansen 1997]. Isto corresponde a dizer que um método de busca local inicia-se a partir de uma solução x qualquer e, sistematicamente, explora a vizinhança desta solução, representada por $\mathcal{N}(x)$, de maneira a encontrar uma solução de melhor custo x' , sendo esta um ótimo local em relação a esta vizinhança.

Uma vizinhança para uma solução x , por exemplo, poderá ser o conjunto de todas as possíveis soluções obtidas a partir de x por meio da complementação de qualquer uma de suas componentes. Obviamente este tipo de vizinhança somente é válido quando as componentes de x , $x_i (i = 1, 2, \dots, N)$, possuírem valores binários ou no intervalo $[0, 1]$. O procedimento de busca local deverá ser encerrado a partir do momento em que tal tipo de movimentação não permita melhorar o custo da solução atual, isto é, explorar todas as soluções que podem ser geradas pelo tipo de movimentação definido para esta vizinhança.

No entanto, em uma metaheurística que considera uma única estrutura de vizinhança em seu algoritmo de busca local, na qual apenas um tipo de movimentação é considerada, haverá mais chances de se ficar preso em um ótimo local. O algoritmo VNS, diferentemente, utiliza-se de um conjunto finito de estruturas de vizinhanças pré-selecionadas, isto é, \mathcal{N}_t , em que $t = t_{min}, \dots, t_{max}$. Para esta definição, $\mathcal{N}_t(x')$ representa o conjunto de todas as soluções contidas na t -ésima vizinhança de x' [Hansen & Mladenović 1997]. O Algoritmo 1 apresenta a estrutura básica de um algoritmo VNS.

Ao se formular um algoritmo VNS, como o descrito no Algoritmo 1, deve-se considerar as seguintes questões:

- Quais e quantos tipos de estruturas de vizinhança \mathcal{N}_t deverão ser utilizadas?
- Qual deverá ser a ordem de exploração destas estruturas de vizinhança?
- Qual estratégia de busca deverá ser utilizada em cada uma destas estruturas de vizinhança?

Algoritmo 1 Estrutura de um algoritmo VNS

- 1: *Inicialização:*
 - 2: *Selecione* um conjunto de estruturas de vizinhanças \mathcal{N}_t , para $t = t_{min}, \dots, t_{max}$, que serão utilizadas na busca local;
 - 3: *Encontre* uma solução inicial x ;
 - 4: **repita**
 - 5: $t \leftarrow t_{min}$
 - 6: **repita**
 - 7: *Perturbação:* Gere um ponto x' aleatoriamente a partir da t -ésima vizinhança de x (isto é, $x' \in \mathcal{N}_t(x)$);
 - 8: *Busca local:* Aplique algum método de busca local utilizando x' como solução inicial; defina como x'' o novo ótimo local obtido;
 - 9: *Atualize ou não:* Se o ótimo local x'' é melhor que o atual x , atualize a solução ($x \leftarrow x''$), e prossiga a busca para $\mathcal{N}_{min}(t \leftarrow t_{min})$; caso contrário, defina $t \leftarrow t + t_{step}$.
 - 10: **até que** $t = t_{max}$
 - 11: **até que** critério de parada atingido
-

As respostas às questões aqui destacadas não são definitivas, pois, a escolha das estruturas de vizinhanças, bem como a ordem de sua aplicação dependerá do problema que se quer tratar. Desta forma, para cada problema deve-se verificar quais estratégias e arranjos são mais eficientes. Esta verificação, por sua vez, é geralmente empírica.

O algoritmo VNS, em sua essência, explora esta ideia de troca estocástica de vizinhança durante a busca por um ótimo global com base nas observações apresentadas a seguir, sendo que a última destas é totalmente empírica, implicando que um ótimo local poderá fornecer informações úteis sobre o ótimo global do problema: [Hansen & Mladenović 2008a]:

- **Observação 1:** um mínimo local relativo à uma vizinhança não necessariamente é um mínimo local em relação à outra;
- **Observação 2:** um mínimo global é um mínimo local em relação a todas as possíveis vizinhanças;
- **Observação 3:** um mínimo local com respeito a uma, ou várias vizinhanças, está normalmente próximo a outro mínimo local;

O funcionamento do algoritmo VNS, descrito no Algoritmo 1, por sua vez, pode ser resumido da seguinte forma: a fim de encontrar uma melhor solução para o problema em questão, uma solução x' é obtida aleatoriamente a partir da t -ésima estrutura de vizinhança, isto é, a estrutura considerada em dado instante de execução do algoritmo VNS. A partir disto, efetua-se o processo de busca local definido para esta vizinhança $\mathcal{N}_t(x')$, exaurindo-se todas as possíveis soluções dentro desta vizinhança até que um ótimo local x'' seja encontrado.

Se o custo desta nova solução x'' , ou seja $f(x'')$ for melhor do que o custo $f(x)$ da melhor solução x conhecida até o momento, atualiza-se x ($x \leftarrow x''$), redefine-se t como $t \leftarrow t_{min}$ e reinicia-se todo o processo. No entanto, caso a nova solução encontrada tiver um custo pior do que a melhor solução tida até o momento, atualiza-se t como $t \leftarrow t + t_{step}$, isto é, efetua-se o processo de busca local a partir da próxima estrutura de vizinhança pré-definida, considerando uma solução inicial x' obtida aleatoriamente para esta estrutura. O algoritmo como um todo deverá ser encerrado assim que algum critério de parada for atingido, como por exemplo o número total de iterações ou o tempo total de execução.

Diferentemente de outras metaheurísticas, como por exemplo, algoritmos genéticos, busca tabu ou colônia de formigas, o número reduzido de parâmetros (t_{min} , t_{step} e t_{max}) necessários ao VNS é um diferencial. De acordo com Hansen et al. (2009) e Hansen & Mladenović (2008b), o algoritmo é de simples ajuste para vários tipos de problema. Mais detalhes sobre o VNS podem ser encontrados em [Hansen & Mladenović 1997, Hansen & Mladenović 2001, Hansen & Mladenović 2008b, Hansen et al. 2009]. A próxima Seção deste texto descreve o VNS-PPMHLP, algoritmo proposto neste trabalho para resolver o problema das p -medianas heterogêneo livre de penalidade.

3.3 VNS-PPMHLP: formulação

A seção anterior deste texto descreveu os principais elementos constituintes do algoritmo VNS em sua versão clássica ao apresentar o Algoritmo 1. Eram eles: (a) inicialização, (b) perturbação, (c) busca local. Todos estes elementos estão presentes no VNS-PPMHLP. No entanto, algumas especializações são propostas, obtendo-se a formulação de um algoritmo específico para o PPMHLP. As próximas subseções deste texto explicam em detalhes a estruturação de cada um destes elementos.

Inicialização

O algoritmo VNS, assim como as demais metaheurísticas, necessita de uma solução inicial para efetuar sua busca por um ótimo global. Esta solução inicial pode ser fornecida pelo usuário ou mesmo por uma heurística construtiva. Tal heurística pode ser, por exemplo, um algoritmo determinístico. O Algoritmo 2 descreve o pseudo-código do método proposto para a obtenção da solução inicial para o VNS-PPMHLP.

Algoritmo 2 VNS-PPMHLP: heurística construtiva

- 1: Calcule uma matriz $\mathcal{F} = (f_{ab})$ com dimensão $I \times I$ de forma que f_{ab} é a norma de Frobenius de $D_{W^a} - D_{W^b}$;
 - 2: Resolva o problema das p -medianas para $p = G$ usando a matriz \mathcal{F} como entrada;
 - 3: **para** $i = 1, \dots, I$ **faça**
 - 4: Defina $p^{ig} = 1$ se a matriz de dissimilaridade do indivíduo i é atribuída à g -ésima mediana; $p^{ig} = 0$ caso contrário;
 - 5: **fim para**
 - 6: **para** $g = 1, \dots, G$ **faça**
 - 7: Resolva cada um dos subproblemas $M_g(p)$;
 - 8: **fim para**
-

O algoritmo descrito primeiro resolve o problema de atribuir os I indivíduos em G segmentos, isto é, encontrar os valores iniciais para as variáveis p do PPMHLP. Considera-se nesta atribuição uma matriz de distâncias $\mathcal{F} = [f_{ab}]_{I \times I}$, em que f_{ab} representa o grau de similaridade entre o indivíduo a ($a = 1, 2, \dots, I$) e o indivíduo b ($b = 1, 2, \dots, I$). O referido grau de similaridade entre estes dois indivíduos é calculado por meio da norma de Frobenius de suas matrizes de dissimilaridades, obtidas a partir da tarefa de triagem para cada um destes indivíduos, isto é $f_{ab} = \|D_{W^a} - D_{W^b}\|_{\mathcal{F}}$.

Uma vez que os valores das variáveis p do PPMHLP sejam conhecidos, pode-se obter uma estrutura de categorias para cada um dos G segmentos aos quais os indivíduos foram atribuídos, ação que corresponde a encontrar os valores das variáveis e . Isto é possível, pois, uma vez que os valores das variáveis p sejam conhecidos, pode-se calcular o número total de categorias (medianas) que cada segmento deverá ter. Formalmente, para se encontrar a solução inicial do PPMHLP, deve-se resolver os G subproblemas $M_g(p)$ ($g = 1, \dots, G$), dados por:

$$\text{Minimize } M_g(p) = \sum_{j=1}^J \sum_{k=1}^K \bar{d}_{jk}^g e_{jk}^g, \quad (3.3)$$

sujeito a

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (3.4)$$

$$\sum_{j=1}^J e_{jj}^g = \lfloor \Omega_g \rfloor, \quad (3.5)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (3.6)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (3.7)$$

em que $\bar{d}_{jk}^g = \sum_{i=1}^I d_{jk}^i p^{ig}$, e $\Omega_g = \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}}$.

Assim como no processo de obtenção dos valores iniciais para as variáveis p , resolver os subproblemas $M_g(p)$, dados em (3.3–3.7), corresponde a encontrar uma solução de boa qualidade, ou mesmo ótima, para o problema das p -medianas. Este problema apesar de ser NP-árduo, como já descrito neste trabalho, pode ser resolvido de forma ótima para os casos em que os valores de J não são demasiadamente grandes, como por exemplo, $J = 30$. Neste trabalho, este é o máximo valor considerado.

No entanto, optou-se resolver o problema das p -medianas por meio do algoritmo POPSTAR [Resende & Werneck 2004], isto para todas as vezes em que esta ação se faz necessária, pois testes realizados considerando-se o referido número de objetos mostraram que este algoritmo é capaz de encontrar a solução ótima do problema na maioria dos casos, além de demandar menor tempo de execução, se comparado ao *solver* CPLEX 12.5.

Perturbação (*Shaking*)

O algoritmo VNS-PPMHLP implementa seu componente de perturbação, mais comumente conhecido pelo termo em inglês *shaking*, utilizando-se de movimentos aleatórios que consideram todas as possíveis formas de se remover um indivíduo i de um dos segmentos g e alocá-lo a outro segmento g' . Ou seja, este movimento é aplicado apenas às variáveis p do modelo do PPMHLP.

Este tipo de movimentação considera o valor do parâmetro t , que determina quantos indivíduos são movidos para outros segmentos. Por exemplo, se $t = 2$, logo 2 indivíduos são removidos de seu segmento atual e alocados a outros segmentos de forma aleatória; se $t = 3$, então esta ação é realizada considerando-se 3 indivíduos, e assim sucessivamente.

Busca local

Dada uma solução existente e viável para o PPMHLP, torna-se necessário aplicar uma busca local à esta solução de forma a encontrar um ótimo local em sua vizinhança. Desenvolveu-se a partir desta necessidade uma busca local baseada na abordagem *Variable Neighborhood Descent* (VND), também conhecida como algoritmo de Descida em Vizinhança Variada. Esta abordagem generaliza as observações 1–3, mencionadas anteriormente na Seção 3.2, para algoritmos de busca local.

Hansen et al. (2010) descrevem que o VND é obtido quando há a troca determinística de vizinhanças durante a execução da busca local. O Algoritmo 3 apresenta sua estrutura básica. Segundo os autores, a maior parte dos procedimentos de busca local utilizam, em sua fase de descida, um número muito restrito de vizinhanças, como por exemplo $t_{max} \leq 2$. No entanto, para o VND, dada uma solução inicial x' , considera-se que o mínimo local x'' a ser encontrado durante a busca local deverá ser um mínimo local em relação à todas as t_{max} estruturas de vizinhanças. Isto aumenta as chances de que o ótimo encontrado seja o global, diferentemente dos casos em que se considera apenas uma estrutura de vizinhança.

Adicionalmente, em um VND, as estruturas de vizinhanças a serem exploradas são aninhadas. Por exemplo, considerando-se a descrição dada no Algoritmo 3, suponha que $t_{max} = 3$ e que o VND esteja sendo aplicado como busca local em um algoritmo VNS. Para tal, dada uma solução x' , deve-se explorar a estrutura de vizinhança $\mathcal{N}_1(x')$. Ao se encontrar nesta busca um ótimo local x'' , deve-se avaliar se este ótimo local possui um custo melhor do que a solução inicial x' dada. Se o custo da nova solução for melhor, então define-se $t = t_{min}$ e $x' \leftarrow x''$, reiniciando-se o processo. Caso contrário, explora-se a vizinhança $\mathcal{N}_2(x')$, buscando-se um novo ótimo local x'' .

Encontrando-se este novo ótimo local, verifica-se se o custo desta nova solução x'' é melhor do que o custo da solução inicial x' . Em caso positivo, define-se $t = t_{min}$ e $x' \leftarrow x''$. Logo, explora-se novamente a vizinhança $\mathcal{N}_1(x')$. Caso contrário, deve-se explorar a vizinhança $\mathcal{N}_3(x')$. Como esta é a última estrutura de vizinhança do VND descrito como exemplo, caso o ótimo local x'' não possua um custo melhor do que a solução inicial x' dada, deve-se prosseguir para a próxima iteração do algoritmo VNS, aplicando-se novamente o algoritmo de perturbação à melhor solução tida até o momento, para então novamente se executar o VND. No entanto, se x'' tem melhor custo que x' , deve definir x' como $x' \leftarrow x''$ e $t \leftarrow t_{min}$, explorando-se novamente a primeira estrutura de vizinhança.

Conceitualmente, quando um algoritmo VND é utilizado como algoritmo de busca local em um VNS, o algoritmo resultante é chamado de Busca em Vizinhança Variada Generalizada, do inglês *Generalized Variable Neighborhood* (GVNS) [Brimberg et al.

2000, Hansen & Mladenović 2001].

Algoritmo 3 Estrutura de um VND

```

1: Entrada: uma solução  $x'$  e um conjunto de estruturas de vizinhanças  $\mathcal{N}_t$ , para  $t = 1, \dots, t_{max}$ 
2:  $t \leftarrow t_{min}$ 
3: repita
4:    $x'' \leftarrow$  melhor solução em  $\mathcal{N}_t(x')$ ;
5:   Se  $f(x') > f(x'')$  faça  $x' \leftarrow x''$  e  $t \leftarrow t_{min}$ ; caso contrário  $t \leftarrow t + 1$ ;
6: até que  $t \geq t_{max}$ 
  
```

Para o algoritmo VNS-PPMHLP, otimiza-se a função objetivo por meio de três estruturas de vizinhança distintas ($t_{max} = 3$), considerando-se a estrutura definida para um GVNS. Estas três estruturas de vizinhança são sintetizadas nos itens que seguem:

- *Vizinhança 1*: encontrar as medianas para cada segmento g , condicional à pertinência dos indivíduos nestes segmentos;
- *Vizinhança 2*: redefinir a pertinência dos indivíduos para cada segmento, considerando-se as medianas e objetos a elas associados como já conhecidos;
- *Vizinhança 3*: aumentar o número de medianas da solução atual redefinindo a pertinência dos indivíduos em cada um dos segmentos.

As próximas subseções deste texto descrevem de forma detalhada cada uma destas três vizinhanças listadas.

Vizinhança 1

A primeira estrutura de vizinhança \mathcal{N}_1 para o Algoritmo 3 resolve o problema (3.3–3.7) para cada um dos segmentos modificados pelo processo de perturbação. Isto corresponde à identificar a melhor estrutura de categorias, ou as medianas, para cada um dos segmentos g . Este processo considera que a pertinência de cada indivíduo i é conhecida, portanto o número total de medianas em cada segmento g pode ser facilmente calculado.

Em virtude desta vizinhança ser a que mais vezes é explorada durante a execução do PPMHLP, $t = 1$, optou-se pela utilização do POPSTAR, algoritmo implementado por Resende & Werneck (2004), que resolve o problema das p -medianas clássico. Isto permitiu acelerar a estimação do resultado. Uma primeira experimentação via *branch-and-bound* para esta etapa mostrou-se menos eficiente em tempos de execução.

Empiricamente, constatou-se que o POPSTAR sempre encontrava a solução ótima dos subproblemas para as instâncias utilizadas neste trabalho. Obviamente, nem sempre a solução ótima será encontrada, ou haverá tal garantia, dependendo da instância considerada,

pois trata-se de um método heurístico para o problema. Em razão dos testes realizados, a configuração do algoritmo dada por padrão pelos autores não foi alterada. Dependendo do tamanho das instâncias a serem resolvidas, recomenda-se revisar e adaptar estas configurações.

Vizinhança 2

A segunda estrutura de vizinhança \mathcal{N}_2 considera temporariamente que a estrutura de categorias é conhecida, isto é, os valores das variáveis e , recuperados na vizinhança \mathcal{N}_1 , são utilizados como parâmetros. A melhora da função objetivo, então, se dá por meio da reatribuição ótima dos indivíduos i a cada um dos segmentos g . Obtém-se desta maneira os melhores valores para as variáveis p . Para que isto seja possível, resolve-se o seguinte problema de otimização:

$$\text{Minimize } W(e) = \sum_{i=1}^I \sum_{g=1}^G p^{ig} \tilde{d}^{ig}, \quad (3.8)$$

sujeito a

$$\frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}} \geq \omega_g, \quad \forall g = 1, \dots, G, \quad (3.9)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (3.10)$$

$$p^{ig} \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad (3.11)$$

em que $\tilde{d}^{ig} = \sum_{j=1}^J \sum_{k=1}^J d_{jk}^i e_{jk}^g$, e $\omega_g = \sum_{j=1}^J e_{jj}^g$. O problema (3.8–3.11) é binário, podendo requerer a resolução de alguns nós via algoritmos de *branch-and-bound*. Utilizou-se nesta etapa o *solver* CPLEX 12.5. Empiricamente, constatou-se que na maioria dos casos este subproblema é resolvido ainda no nó raiz e a solução dada é ótima.

No entanto, como esta resolução pode, em alguns raros casos observados, demandar considerável tempo de computação, um limite de tempo foi imposto ao *solver* (um segundo). Se uma solução viável for encontrada, respeitando este limite de tempo, esta é retornada. Caso contrário, o processo de busca segue com a solução atual x . Obviamente um aumento no tempo de busca pode influenciar na qualidade das soluções encontradas nesta etapa, porém os experimentos realizados mostraram que aumentar o tempo de processamento não produz uma melhora significativa.

Vizinhança 3

A terceira estrutura de vizinhança \mathcal{N}_3 considerada na busca local do VNS-PPMHLP é

um híbrido entre a vizinhança \mathcal{N}_1 e a vizinhança \mathcal{N}_2 , pois consiste na resolução alternada de ambas com algumas modificações, a fim de que se otimize o número de medianas em cada um dos segmentos g . Como consequência deste processo, as estruturas de categorias e a alocação dos indivíduos recuperadas em cada segmento também é otimizada. Esta etapa do processo de busca local é apresentada no Algoritmo 4.

Para ilustrar o funcionamento da vizinhança \mathcal{N}_3 , considere que o processo descrito a seguir é executado para todo segmento g^* ($g^* = 1, 2, \dots, G$) de maneira sequencial. Desta forma, primeiramente, uma solução (p^{best}, e^{best}) é inicializada com os valores da melhor solução conhecida até o momento para o PPMHL, isto é, $(p^{best}, e^{best}) \leftarrow (p, e)$.

A partir desta inicialização, resolve-se o problema $M_{g^*}(p)$, dado em (3.3–3.7). No entanto, deve-se substituir o parâmetro Ω_{g^*} deste modelo por $\Omega_{g^*} + 1$. Esta modificação significa que o número de medianas do segmento g^* deve ser acrescido em uma unidade, e portanto a estrutura de categorias deste segmento será redefinida, isto é, os valores das variáveis e^{g^*} . Esta estratégia sempre ocasionará melhora no custo da função objetivo, pois o custo relativo ao segmento g^* será reduzido, supondo que o número de medianas é menor que o número de objetos.

Esta redução ocorre em função de que um objeto a mais, obrigatoriamente, será definido como mediana dentro do segmento g^* . Como este objeto estará associado a si mesmo, ligação em que o custo é nulo, a melhora na função objetivo sempre ocorrerá. No entanto, esta nova solução gerada para as variáveis e^* poderá tornar a solução (p, e) inviável para o PPMHL, pois poderá não existir um conjunto de valores para as variáveis p que faça com que as restrições dadas a seguir sejam válidas para todos os segmentos g :

$$\sum_{j=1}^J e_{jj}^g = \left\lfloor \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}} \right\rfloor, \quad \forall g = 1, \dots, G, \quad (3.12)$$

de modo a acomodar o aumento no número de medianas em g^* .

Desta forma, o problema $W(e)$, dado em (3.8–3.11), deverá ser resolvido. Por meio deste, busca-se reatribuir os indivíduos entre todos os segmentos. A resolução deste problema retornará uma solução viável caso seja possível reatribuir todos os indivíduos de forma que as restrições em (3.12) sejam atendidas, isto para todo segmento g . Caso isto ocorra, tem-se uma nova solução (p, e) de melhor custo e viável para o PPMHL, e portanto deve-se atualizar (p^{best}, e^{best}) como $(p^{best}, e^{best}) \leftarrow (p, e)$.

Caso a solução retornada pela resolução do problema $W(e)$ seja inviável, isto significa que não existe um conjunto de valores para as variáveis p que satisfaça o conjunto de restrições em (3.12). Desta forma, descarta-se esta nova solução (p, e) . Ao se encer-

rar o processo aqui descrito, deve-se repeti-lo para o próximo segmento $g^* + 1$, e assim sucessivamente até que todos os demais segmentos sejam considerados.

Algoritmo 4 VNS-PPMHLP: \mathcal{N}_3

```

1: Entrada: uma solução  $(p, e)$  para o PPMHLP
2:  $(p^{best}, e^{best}) \leftarrow (p, e)$ 
3: para  $g^* = 1, \dots, G$  faça
4:   Minimize  $M_{g^*}(p)$  com  $\Omega_{g^*}$  substituído por  $\Omega_{g^*} + 1$ ;
5:   Minimize  $W(e)$ 
6:   se  $W(e) = +\infty$  ou o custo de  $(p, e)$  é maior que o custo de  $(p^{best}, e^{best})$  então
7:      $(p, e) \leftarrow (p^{best}, e^{best})$ ;
8:   senão
9:      $(p^{best}, e^{best}) \leftarrow (p, e)$ ;
10:  fim se
11: fim para
12: retorne  $(p^{best}, e^{best})$ 

```

Considerações sobre o VNS-PPMHLP

O fato de o problema das p -medianas ser NP-árduo [Kariv & Hakimi 1979], e que ele ainda está presente no VNS-PPMHLP, mesmo após o processo de reformulação apresentado no Capítulo 2, contribui para que este seja de complexa resolução. A existência de variáveis binárias no problema é outro entrave ao uso de algoritmos exatos tradicionais.

Como mostrado nos experimentos computacionais relatados na Seção 2.4, tornou-se possível apenas a obtenção de limitantes inferiores para um conjunto muito reduzido de instâncias, além de soluções inteiras de qualidade questionável. Tudo isto, obviamente, a um custo computacional elevado e tempos de execução muito grandes. Um outro experimento realizado mostrou que nem uma execução de três semanas foi capaz de resolver uma instância real do problema de forma satisfatória. Esta instância será descrita ao final deste capítulo.

Justifica-se, portanto, o desenvolvimento do algoritmo proposto neste Capítulo, o VNS-PPMHLP. Este incorpora diferentes estruturas de vizinhança, valendo-se do conceito do VND, acelerando assim o processo de otimização e tornando-o viável mesmo para simples computadores de mesa com poder de processamento e memória significativamente limitados. Os resultados computacionais que serão apresentados a seguir comprovam esta afirmação.

3.4 VNS-PPMHLP: resultados computacionais

Nesta seção o algoritmo VNS proposto para a resolução do problema das p -medianas heterogêneo livre de penalidade, descrito na Seção 3.3, será avaliado de modo a verificar não só sua eficiência e precisão, mas também validar o novo modelo dado ao problema das p -medianas heterogêneo de [Blanchard et al. 2012]. Primeiramente, o algoritmo é avaliado em relação à sua eficiência computacional, comparando-se as soluções obtidas por ele àquelas obtidas por meio do *solver* CPLEX 12.5. Para esta comparação, os resultados computacionais de ambas as formulações, PPMHLP1 e PPMHLP2, serão utilizadas.

Posteriormente, os resultados obtidos pelo algoritmo descrito serão comparados aos resultados dados pelo algoritmo de Blanchard et al. (2012), o VNS-PPMH (Algoritmo de Busca em Vizinhança Variada para o Problema das p -Medianas Heterogêneo), que resolve o problema original dos autores. Dado que os referidos algoritmos resolvem modelos diferentes, não é possível compará-los em relação aos custos das funções objetivo. Desta forma, optou-se pela utilização do Índice de Rand Ajustado (*Adjusted Rand Index* - ARI) [Hubert & Arabie 1985], que permite mensurar a precisão dos algoritmos ao recuperar a informação contida nas instâncias em relação às estruturas de categorias e à segmentação dos indivíduos, ou seja, a precisão ao recuperar os valores das variáveis p e e .

Para tal, serão consideradas as instâncias geradas pela Simulação de Monte Carlo, descritas na Seção 2.4 deste texto. A fim de recapitular detalhes da realização dos experimentos computacionais que são apresentados neste trabalho, as configurações aplicadas aos algoritmos VNS-PPMHLP e VNS-PPMH são listadas:

- Realizou-se 10 execuções para cada algoritmo;
- Todas as execuções tiveram um tempo limite de 10 minutos;
- Os experimentos foram realizados em um computador com 12 processadores Intel(R) Xeon(R) CPU X5650 de 2.67GHz e 62GB de memória RAM;
- Cada execução, para cada um dos algoritmos, só utilizou um processador (*single thread*).

Ambos os algoritmos, em sua essência, não são paralelos. Porém, ambos utilizam o *solver* CPLEX 12.5 em alguns de seus subproblemas. Este *solver*, por sua vez, pode utilizar todos os processadores disponíveis ao mesmo tempo, haja visto que faz uso de paralelismo. Neste caso, limitou-se o número de processadores que este programa poderia utilizar a apenas um.

As configurações supracitadas foram utilizadas na resolução tanto das instâncias geradas por simulação quanto para instâncias reais, as quais representam um breve exemplo

de aplicação do PPMHLP. Este exemplo é mostrado ao final deste capítulo. Um importante detalhe acerca dos algoritmos VNS-PPMHLP e VNS-PPMH é que ambos puderam executar, para todas as instâncias, reais ou simuladas, com no máximo 2GB de memória. Em termos práticos, isto quer dizer que podem ser aplicados mesmo com recursos computacionais limitados.

3.4.1 Análise de estabilidade e desempenho

Avalia-se, nesta seção, se o VNS-PPMHLP é capaz de recuperar as mesmas soluções sempre que executado para o mesmo conjunto de dados, apesar de não ser um algoritmo determinístico. Avalia-se também se este algoritmo resolve o PPMHLP de forma mais eficiente que os algoritmos exatos utilizados, para os quais os resultados computacionais foram apresentados na Seção 2.4. Entenda-se por eficiente, neste contexto, capaz de obter soluções de melhor custo obtidas com menor esforço computacional.

A primeira análise a ser apresentada é acerca da estabilidade e confiabilidade do algoritmo. A Tabela 3.1 apresenta os custos obtidos a partir das 10 execuções do VNS-PPMHLP para cada uma das 27 instâncias geradas por simulação. Como esta tabela tem uma estrutura de simples compreensão, suas colunas não serão descritas detalhadamente, exceto as três últimas. A coluna "Melhor Custo" apresenta o melhor valor obtido para a função objetivo a partir das 10 réplicas do experimento. A coluna "Pior Custo", por sua vez, o custo mais elevado. A coluna "Dif.%" apresenta a diferença relativa entre estas duas.

Nota-se, ao observar a referida tabela, que a maior parte das instâncias (20 ao todo) apresentam uma diferença relativa inferior a 0,5%. Dentre estas, 15 apresentam diferença nula. Os piores valores para esta métrica, no entanto, são observados para as instâncias maiores. Por exemplo, o pior valor observado é 5,10%, relativo à Instância 8. Esta é resolvida para 10 segmentos ($G = 10$) e apresenta uma composição heterogênea em relação à estrutura de categorias.

O segundo pior resultado é relativo à Instância 27, com uma distância relativa de 2,02%. Esta instância também é resolvida para $G = 10$. Embora não possua variação no número de categorias definidas na simulação, o que torna a instância mais difícil de ser resolvida, esta possui a máxima quantidade de indivíduos ($I = 450$) e máxima quantidade de objetos considerados na simulação ($J = 30$).

As Instâncias 5 e 24, apresentam uma variação relativamente menor, 1,97% e 0,97%, respectivamente. No entanto, as características destas instâncias são semelhantes àquelas cujo os resultados são piores (Instâncias 8 e 27), descritas anteriormente. Ambas conside-

ram resolver o problema para $G = 10$, com $I = 300$ e $I = 450$. Estes resultados sugerem, em um primeiro momento, que o VNS-PPMHLP é sensível à variação no número de segmentos para valores elevados de G , havendo uma redução em seu desempenho. Desta forma, dependendo da aplicação dada ao VNS-PPMHLP, deve-se considerar reconfigurar o algoritmo e efetuar diferentes testes, para que as soluções obtidas sejam mais similares entre si. No entanto, o algoritmo mostrou-se robusto para as instâncias com $G \leq 6$.

Em termos práticos, pequenas variações na função objetivo, como as aqui observadas, são toleráveis. Para fins de análise em casos reais, obviamente, a solução de menor custo deverá ser utilizada. Esta prática foi adotada por Blanchard et al. (2012) em seu estudo, o qual aplicou o PPMH à segmentação de consumidores, isto é, cenário em que não se minimiza um custo monetário. Na ocasião, seu algoritmo também apresentou variações em relação ao custo das soluções obtidas para cada resolução, inclusive para instâncias de menor porte que as apresentadas nesta seção.

Inst.	Custo obtido em cada uma das 10 execuções										Melhor Custo	Pior Custo	Dif. %
	1	2	3	4	5	6	7	8	9	10			
1	3206.41	3207.18	3207.78	3207.78	3208.14	3208.80	3209.06	3209.06	3209.21	3209.21	3206.41	3209.21	0.09%
2	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	2388.25	0.00%
3	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	4604.21	0.00%
4	1870.00	1870.03	1870.06	1870.06	1870.07	1870.07	1870.07	1870.07	1870.07	1870.07	1870.00	1870.07	0.00%
5	3756.43	3758.64	3759.05	3759.96	3760.22	3762.38	3825.26	3825.67	3828.48	3831.81	3756.43	3831.81	1.97%
6	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	1525.20	0.00%
7	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	2651.01	0.00%
8	1554.33	1564.33	1577.33	1577.33	1579.33	1580.33	1585.67	1585.67	1588.33	1637.83	1554.33	1637.83	5.10%
9	7366.17	7366.53	7366.54	7367.14	7372.47	7375.29	7402.47	7403.61	7403.68	7411.27	7366.17	7411.27	0.61%
10	5677.51	5678.73	5679.47	5679.56	5680.63	5681.73	5682.21	5682.52	5684.53	5686.73	5677.51	5686.73	0.16%
11	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	2835.41	0.00%
12	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	3749.99	0.00%
13	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	3562.48	0.00%
14	3082.00	3082.17	3082.17	3082.17	3082.17	3082.17	3082.17	3084.50	3086.00	3093.17	3082.00	3093.17	0.36%
15	5760.95	5761.13	5761.13	5761.13	5761.13	5761.13	5761.13	5761.13	5761.13	5761.13	5760.95	5761.13	0.00%
16	9869.40	9870.70	9870.70	9870.70	9870.70	9870.70	9870.70	9870.70	9870.70	9870.70	9869.40	9870.70	0.01%
17	3125.28	3128.04	3128.47	3128.47	3130.20	3143.69	3151.23	3152.02	3152.02	3152.02	3125.28	3152.02	0.85%
18	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	6497.31	0.00%
19	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	1206.01	0.00%
20	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	10935.00	0.00%
21	3593.91	3594.32	3594.68	3594.75	3594.95	3595.03	3595.15	3595.15	3595.31	3595.73	3593.91	3595.73	0.05%
22	4611.44	4611.48	4611.58	4611.71	4611.71	4611.71	4611.71	4611.71	4611.71	4611.71	4611.44	4611.71	0.01%
23	5673.25	5674.42	5674.74	5675.01	5675.03	5675.31	5675.42	5675.46	5676.09	5676.09	5673.25	5676.09	0.05%
24	3761.34	3763.88	3764.13	3764.77	3770.30	3785.21	3791.17	3794.19	3796.54	3798.10	3761.34	3798.10	0.97%
25	1247.64	1247.73	1247.73	1247.73	1247.73	1247.73	1247.73	1247.73	1248.01	1248.43	1247.64	1248.43	0.06%
26	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	1874.99	0.00%
27	8799.20	8846.40	8850.00	8856.60	8901.60	8902.00	8906.00	8911.60	8915.00	8980.80	8799.20	8980.80	2.02%

Tabela 3.1: VNS-PPMHLP: custos das soluções obtidas para 10 execuções

O próximo fator a ser analisado é a eficiência do algoritmo VNS-PPMHLP e seu método de busca local na obtenção de soluções de melhor custo em relação aos algoritmos exatos utilizados. Primeiramente, apresenta-se na Tabela 3.2 os custos das soluções iniciais obtidas para cada uma das instâncias simuladas por meio da heurística construtiva, assim como o custo da solução final obtida após a aplicação do VNS. Neste caso, compara-se o custo inicial à solução de melhor custo dentre as 10 execuções.

Esta análise será um tanto breve, pois para algumas instâncias simuladas a solução inicial dada pela heurística construtiva é igual à solução final. Isto é, não existe melhora. Isto é bem visível nos casos em que o nível de perturbação adicionado às matrizes de dissimilaridades e ao número de pilhas é relativamente baixo ou nulo. No entanto, pode-se observar na Tabela 3.2 que o VNS-PPMHLP produz melhora nas soluções iniciais dadas para as instâncias cujo nível de ruído acrescido às matrizes de dissimilaridades e ao número de pilhas é maior. No total, melhoras são observadas em 70% das soluções iniciais para estas instâncias.

Relembra-se aqui que a heurística construtiva formulada para o VNS-PPMHLP calcula a norma de Frobenius entre as matrizes de dissimilaridade D_{wi} de todos os indivíduos i , par-a-par. Logo, se o nível de ruído adicionado à estas matrizes é zero, a norma será zero para os indivíduos com as mesmas estruturas de categorias. Portanto, indivíduos com as mesmas estrutura de categorias serão alocados ao mesmo segmento.

A eficiência do VNS-PPMHLP e sua busca local poderá ser notada com maior impacto quando de sua aplicação a um conjunto de dados reais (ver Seção 3.4.4), pois nestes casos a solução inicial tende a ser mais distante da solução ótima.

Por fim, apresenta-se na Tabela 3.3 os custos das melhores soluções encontradas por meio do VNS-PPMHLP em relação às instâncias geradas pela Simulação de Monte Carlo. Apresenta-se, novamente, as soluções dadas pelo *solver* CPLEX 12.5 para as formulações PPMHLP1 e PPMHLP2, no intuito de se comparar estes resultados aos do algoritmo proposto.

Para esta tabela, os valores apresentados para as colunas GAP referem-se à distância, em porcentagem, entre o custo da melhor solução inteira obtida, para todos os casos, em relação ao menor custo da relaxação obtido por meio da formulação PPMHLP2, considerando todos os nós da árvore de resolução do algoritmo *branch-and-bound* (melhor limitante inferior). Compara-se também os piores custos obtidos a partir do VNS-PPMHLP durante as dez execuções.

Estes resultados mostram claramente a superioridade do VNS-PPMHLP, pois para 25 instâncias dentre as 27 o algoritmo obtém soluções de melhor custo, considerando-se suas melhores soluções. A primeira exceção é observada em relação à Instância 7, para a

qual as três abordagens obtêm soluções de custos iguais. A segunda exceção é observada para a Instância 26, onde todos os métodos obtiveram a solução ótima do problema. No entanto, apenas a formulação PPMHLP2 permite a prova desta otimalidade, haja visto que a formulação PPMHLP1 não fornece qualquer limitante inferior.

Apesar de a solução ótima ter sido encontrada nos três casos, destaca-se que este resultado é o de menor relevância para avaliar o desempenho em termos de custo, uma vez que esta instância é a de resolução mais simples. Além de ser a de menor tamanho, não apresenta nenhum grau de perturbação, tanto para o número de pilhas c^i quanto para as matrizes de dissimilaridades D_{wi} . Para esta instância, a heurística construtiva do VNS-PPMHLP fornece a solução ótima.

Um resultado a ser relevado, no entanto, é referente à resolução da Instância 12. Para esta, o VNS-PPMHLP encontrou a solução ótima, haja visto que o custo desta solução é igual ao custo da menor relaxação retornada pela resolução da formulação PPMHLP2 via algoritmos exatos. O *solver*, diferentemente, encontrou uma solução inteira muito distante desta ao resolver a formulação PPMHLP2. O fato interessante a ser observado é que o limitante inferior dado por esta formulação foi obtido no nó raiz de seu algoritmo, sendo que este algoritmo conseguiu resolver apenas este nó em um período de 24h. Isto mostra o quão forte é a formulação obtida para o problema a partir da aplicação das técnicas de relaxação e linearização utilizadas.

Contrariamente, a formulação PPMHLP1, fornece uma solução muito próxima do ótimo global, com um GAP igual a 0,79% em relação ao menor valor de relaxação retornado pelo *solver* ao resolver a formulação PPMHLP2. Atribui-se isto ao fato de que esta formulação, por possuir menor número de variáveis de decisão permitiu ao *solver* resolver uma maior quantidade de nós em seu algoritmo, sendo esta quantidade acima de 30 mil nós.

Finalmente, ao se comparar os custos das piores soluções obtidas pelo algoritmo VNS-PPMHLP dentre as dez execuções, observa-se um resultado similar. O algoritmo VNS-PPMHLP obtém soluções de melhor custo para 24 das 27 instâncias. Neste caso, o algoritmo obtém uma solução de pior custo em relação à formulação PPMHLP1 apenas para a Instância 8. Porém, novamente o VNS-PPMHLP, mesmo considerando-se as soluções de pior custo, obtém as soluções ótimas para as Instâncias 12 e 26.

Conclui-se, a partir dos resultados mostrados nesta seção, que o VNS-PPMHLP é apropriado para a resolução do PPMHLP. No entanto, até este ponto, estende-se esta conclusão apenas em relação ao seu desempenho e estabilidade. Como mostrado, além de obter soluções de melhor ou igual custo para todos os casos considerados, quando comparado ao *solver* CPLEX 12.5, a demanda por poder e tempo de computação do

algoritmo é mínima.

Instância	Detalhes das instâncias simuladas						Custo HC	Custo VNS-PPMHLP
	<i>I</i>	<i>G</i>	<i>J</i>	Categorias	Perturbação (Dissimilaridades)	Perturbação (Pilhas)		
1	150	10	30	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)	3257.66814	3206.40872
2	300	2	18	6	N(0, 0.1)	0	2388.25282	2388.25282
3	450	2	18	50 % 3, 50 % 6	N(0, 0.05)	0	4604.20725	4604.20725
4	150	2	18	3	N(0, 0.05)	N(0, 0.5)	1996.79564	1870.00148
5	450	10	18	6	N(0, 0.05)	N(0, 1)	3936.21208	3756.43118
6	150	10	18	50 % 3, 50 % 6	N(0, 0.05)	0	1525.19737	1525.19737
7	300	2	18	6	0	N(0, 0.5)	2900.01	2651.01098
8	150	10	18	50 % 3, 50 % 6	0	N(0, 1)	1785.66664	1554.33254
9	300	10	30	3	N(0, 0.05)	N(0, 0.5)	7604.89814	7366.17366
10	450	6	18	3	N(0, 0.1)	N(0, 1)	6049.08562	5677.5146
11	150	6	30	6	N(0, 0.1)	0	2835.40836	2835.40836
12	300	10	18	3	0	0	3749.985	3749.985
13	450	10	18	6	N(0, 0.1)	0	3562.48493	3562.48493
14	300	6	18	50 % 3, 50 % 6	0	N(0, 1)	3249.999	3081.99853
15	300	2	30	6	N(0, 0.05)	N(0, 1)	6000.37354	5760.95053
16	450	2	30	50 % 3, 50 % 6	0	N(0, 1)	10192.5	9869.4
17	300	6	18	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)	3441.23912	3125.28194
18	300	6	30	50 % 3, 50 % 6	N(0, 0.05)	0	6497.31133	6497.31133
19	150	6	18	6	0	N(0, 0.5)	1241.6725	1206.00594
20	450	6	30	3	0	0	10935	10935
21	150	2	30	3	N(0, 0.1)	N(0, 1)	3709.92128	3593.90695
22	450	2	18	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)	4929.876	4611.43964
23	450	6	18	3	N(0, 0.05)	N(0, 0.5)	5987.09113	5673.25235
24	300	10	18	3	N(0, 0.1)	N(0, 1)	3945.63625	3761.34159
25	150	6	18	6	N(0, 0.05)	N(0, 1)	1389.30303	1247.64053
26	150	2	18	3	0	0	1874.9925	1874.9925
27	450	10	30	6	0	N(0, 0.5)	9207	8799.2

Tabela 3.2: VNS-PPMHLP: heurística construtiva e resultados finais

Instância	PPMHL P2 (CPLEX)				PPMHL P1 (CPLEX)				VNS-PPMHL P			
	Melhor Custo Solução Inteira	Menor Custo Relaxação	GAP	Nós Explorados	Melhor Custo Solução Inteira	GAP	Nós Explorados	Melhor Custo	GAP	Pior Custo	GAP	
1	-	-	100.00%	-	3507.7327	100.00%	158485	3206.40872	100.00%	3209.21	100.00%	
2	2389.32036	2336.58209	2.21%	8310	2391.94615	2.31%	410248	2388.25282	2.16%	2388.25	2.16%	
3	4607.0683	4398.937717	4.52%	1097	5104.79194	13.83%	330278	4604.20725	4.46%	4604.21	4.46%	
4	1870.07439	1823.609552	2.48%	41186	1871.15249	2.54%	2966838	1870.00148	2.48%	1870.07	2.48%	
5	7353.25648	0	100.00%	1	4900.87911	100.00%	22200	3756.43118	100.00%	3831.81	100.00%	
6	2396.44816	1382.715212	42.30%	276	1762.59574	21.55%	82193	1525.19737	9.34%	1525.20	9.34%	
7	2651.01098	2413.333147	8.97%	3432	2651.01098	8.97%	1756295	2651.01098	8.97%	2651.01	8.97%	
8	1824.16658	1470.83555	19.37%	8	1585.16597	7.21%	84443	1554.33254	5.37%	1637.83	10.20%	
9	8425.27766	0	100.00%	1	7784.44564	100.00%	7443	7366.17366	100.00%	7411.27	100.00%	
10	7267.39799	4995.302978	31.26%	3	5889.9318	15.19%	58635	5677.5146	12.02%	5686.73	12.16%	
11	4235.06645	0	100.00%	1	2958.48186	100.00%	585748	2835.40836	100.00%	2835.41	100.00%	
12	4849.995	3749.985	22.68%	1	3779.9856	0.79%	374311	3749.985	0.00%	3749.99	0.00%	
13	7330.23751	0	100.00%	1	4678.67604	100.00%	21621	3562.48493	100.00%	3562.48	100.00%	
14	4648.16355	2905.83791	37.48%	15	3227.33306	9.96%	354894	3081.99853	5.72%	3093.17	6.06%	
15	6087.37635	5644.11692	7.28%	677	5775.7719	2.28%	105515	5760.95053	2.03%	5761.13	2.03%	
16	10752.5	9670.02	10.07%	29	10204.2	5.23%	53782	9869.4	2.02%	9870.70	2.03%	
17	4127.72272	2671.693225	35.27%	1601	3386.6488	21.11%	90194	3125.28194	14.51%	3152.02	15.24%	
18	8448.21217	0	100.00%	1	7073.90629	100.00%	14741	6497.31133	100.00%	6497.31	100.00%	
19	1269.00531	1190.00595	6.23%	342	1264.67202	5.90%	138893	1206.00594	1.33%	1206.01	1.33%	
20	12645	0	100.00%	1	11435.7	100.00%	16408	10935	100.00%	10935.00	100.00%	
21	3726.85898	3405.44083	8.62%	111	3639.28456	6.43%	8291974	3593.90695	5.24%	3595.73	5.29%	
22	4710.93306	4233.501076	10.13%	1871	4906.24547	13.71%	188348	4611.43964	8.20%	4611.71	8.20%	
23	7262.72169	5312.943074	26.85%	5	5784.20158	8.15%	46099	5673.25235	6.35%	5676.09	6.40%	
24	4834.31262	3212.933781	33.54%	3	3983.76823	19.35%	83792	3761.34159	14.58%	3798.10	15.41%	
25	1460.81456	1141.321417	21.87%	1752	1363.56496	16.30%	147203	1247.64053	8.52%	1248.43	8.58%	
26	1874.9925	1874.9925	0.00%	1	1874.9925	0.00%	3330090	1874.9925	0.00%	1874.99	0.00%	
27	12690	0	100.00%	1	10103.4	100.00%	23794	8799.2	100.00%	8980.80	100.00%	

Tabela 3.3: Comparação entre resultados computacionais obtidos para as formulações PPMHL P1 e PPMHL P2 via *solver* em relação ao VNS-PPMHL P

A próxima seção deste texto, adicionalmente, apresentará uma análise em relação à precisão com que o algoritmo recupera as informações, comparando-o ao algoritmo de Blanchard et al. (2012). Esta recuperação da informação é relativa à recuperação da estrutura de categorias estabelecida pelos indivíduos durante a tarefa de triagem e à alocação destes indivíduos aos diferentes segmentos.

3.4.2 Recuperação da informação

Na Subseção 3.4.1 mostrou-se que o algoritmo VNS-PPMHLP é a melhor alternativa para a resolução do novo modelo proposto neste trabalho, o PPMHLP. Isto devido ao fato de que as formulações PPMHLP1 e PPMHLP2 não puderam ser resolvidas via algoritmos exatos tradicionais em tempo razoável para as instâncias consideradas.

Nesta seção, analisa-se a precisão com que os algoritmos VNS-PPMHLP e VNS-PPMH recuperam a estrutura de categorias previamente definida por meio da Simulação de Monte Carlo, assim como a segmentação correta dos indivíduos. De modo a comparar o VNS-PPMHLP e o VNS-PPMH e seus respectivos problemas, PPMHLP e PPMH, considerou-se que utilizar o ARI [Hubert & Arabie 1985] é apropriado, pois os modelos não são equivalentes em razão de suas funções objetivo e da forma como limitam o número de medianas para cada segmento.

O ARI permite comparar se os valores preditos em ambos os modelos para as variáveis e correspondem aos valores observados W_{jl}^i , que representam em qual categoria l um indivíduo i classificou o objeto j durante a tarefa de triagem. Obviamente os valores para as variáveis e que serão comparados à estrutura de categorias feita pelo indivíduo i estão condicionados à pertinência deste indivíduo em um dos segmentos g , ou seja, o valor de p^{ig} . O ARI tem como resultado um valor cujo teto é 1 (escalar) e este valor indica uma recuperação perfeita da informação.

De forma a ilustrar a obtenção dos valores de entrada considerados no cálculo do ARI para as variáveis e , suponha que um indivíduo i^* construiu uma estrutura de categorias com $c^{i^*} = 3$ pilhas ao analisar $J = 5$ objetos. O resultado de sua classificação é dado pela matriz W^{i^*} apresentada a seguir, na qual cada linha representa um objeto j e sua respectiva pertinência em uma das pilhas l :

$$W^{i^*} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \quad (3.13)$$

Para fins de simplificação, pode-se reescrever este resultado como um vetor $U^{i^*} = [u_j]_J$, no qual cada posição j conterà o número da pilha l em que o objeto j foi colocado. Desta forma, tem-se:

$$U^{i^*} = \begin{bmatrix} 1 & 2 & 3 & 1 & 1 \end{bmatrix}. \quad (3.14)$$

O vetor U^{i^*} representa a estrutura de categorias feita pelo indivíduo i^* durante a tarefa de triagem. Para que se possa calcular o ARI para este indivíduo i^* em relação às variáveis e , deve-se agora considerar outro vetor V^{g^*} , o qual irá representar a estrutura de categorias recuperada para o segmento g^* . Obviamente, para que esta comparação faça sentido, deve-se supor que o indivíduo i^* foi alocado a um segmento g^* , o que implica que $p^{i^*g^*} = 1$.

A partir disto, considera-se que a estrutura de categorias recuperada pelo PPMHLP para o segmento g^* é dada pela matriz $e^{g^*} = [e_{jk}]_{J \times J}$. Para fins de exemplificação, considera-se também que esta matriz contém os seguintes valores:

$$e^{g^*} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.15)$$

Pela definição do modelo do PPMHLP, os valores na diagonal principal desta matriz, quando iguais a 1, indicam que o objeto referente a esta linha e coluna é uma mediana. Portanto, ao se observar a matriz dada em (3.15), vê-se que existem 3 medianas, sendo elas os objetos 1, 2 e 4. Como existem três medianas, existem três categorias às quais os demais objetos foram atribuídos, sendo que o objeto 5 foi atribuído à mediana referente ao objeto 1 e o objeto 3 foi atribuído à mediana representada pelo objeto 4.

Pode-se reescrever a matriz dada em (3.15) como uma nova matriz q^{g^*} , eliminando-se

suas colunas que contenham apenas zeros, ou seja:

$$q^{g^*} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}. \quad (3.16)$$

A partir desta matriz q^{g^*} , então, obtém-se o vetor V^{g^*} :

$$V^{i^*} = \begin{bmatrix} 1 & 2 & 3 & 3 & 1 \end{bmatrix}. \quad (3.17)$$

De posse dos vetores U^{i^*} e V^{g^*} , pode-se facilmente calcular o ARI (ver [Hubert & Arabie 1985]) relativo à comparação entre a estrutura de categorias feita pelo indivíduo i^* na tarefa de triagem e a estrutura de categorias recuperada para o segmento g^* pelos algoritmos VNS-PPMHLP e VNS-PPMH.

Utiliza-se também o ARI para comparar os modelos em relação às variáveis p estimadas com relação aos valores gerados na criação das instâncias. Para todos os casos, variáveis e e variáveis p , o ARI médio é apresentado 1.33 para cada instância. A Tabela 3.4 mostra estes resultados. Como a extração dos valores para o cálculo do ARI em relação às variáveis p é um tanto simples, este processo não será explicado. As subseções a seguir analisam os resultados obtidos.

Variáveis e

Um teste- t para dados pareados em relação às variáveis e mostra que a diferença entre o ARI médio observado para os algoritmos VNS-PPMHLP e VNS-PPMH é significativa. Para o VNS-PPMHLP, tem-se que $\overline{ARI}_{VNS-PPMHLP} = 0,952$ e $DP_{VNS-PPMHLP} = 0,059$. Para o VNS-PPMH, $\overline{ARI}_{VNS-PPMH} = 0,801$ e $DP_{VNS-PPMH} = 0,263$. Para estes valores, a diferença entre as médias é significativa, pois $\text{valor-}t(26) = 3,20$ e $\text{valor-}p < 0,01$. Desta forma-se, mostra-se que o VNS-PPMHLP é mais preciso ao recuperar a estrutura de categorias que o VNS-PPMH.

Adicionalmente, pode-se analisar os mesmos resultados removendo-se as instâncias cujo ARI obtido foi igual a 1 em ambos os algoritmos. Para as instâncias remanescentes, ao todo 18, observa-se que o VNS-PPMHLP supera o VNS-PPMH em 14 casos. Conclui-se novamente, a partir desta observação, que o VNS-PPMHLP demonstra superioridade em relação ao VNS-PPMH na recuperação da estrutura de categorias.

Variáveis p

As variáveis p , por sua vez, indicam a qual segmento g cada um dos I indivíduos pertence. Ambos os modelos apresentam boa performance em relação ao ARI. Para estas variáveis, tem-se $\overline{ARI}_{VNS-PPMHLP} = 0,893$ e $DP_{VNS-PPMHLP} = 0,169$. Para o VNS-PPMH, tem-se $\overline{ARI}_{VNS-PPMH} = 0,851$ e $DP_{VNS-PPMH} = 0,236$. Apesar de o VNS-PPMHLP parecer superior, à primeira vista, a diferença entre estas médias não é significativa, pois um teste- t apresenta como resultados valor- $t(26) = 1,18$ e valor- $p = 0,25$. Portanto, neste quesito os modelos e algoritmos são equivalentes, apresentando o mesmo grau de precisão ao recuperar a segmentação pré-definida no processo de simulação para os indivíduos.

3.4.3 Sensibilidade a diferentes fatores

Para avaliar o quão sensíveis são os algoritmos VNS-PPMHLP e VNS-PPMH, bem como seus respectivos modelos, aos diferentes fatores que caracterizam cada instância gerada pela Simulação de Monte Carlo, fez-se a predição do ARI para cada um dos modelos por meio de regressão linear múltipla. Utilizou-se a codificação destes diversos fatores na forma de variáveis binárias, abordagem mais conhecida na área de estatística pelo nome em inglês *dummy-coded variables*, pois alguns fatores não possuem valores numéricos. Os resultados desta regressão são mostrados na Tabela 3.5 para a predição do ARI em relação às variáveis e e na Tabela 3.6 para as variáveis p .

Para as referidas tabelas, cada linha apresenta os coeficientes beta para os fatores utilizados como variáveis independentes, juntamente com o nível de significância de cada um dos fatores. Caso o fator se mostre significativo, o modelo em questão é sensível aquela característica, especialmente. Caso contrário, mostra-se que o modelo é robusto em relação ao fator considerado. Os valores identificados como mais significantes em ambas as tabelas estão marcados com um asterisco em sua respectiva linha.

Primeiramente, examina-se o grau de sensibilidade dos algoritmos e modelos aos diversos fatores da simulação, ponderando-se os resultados da regressão para o ARI em relação às variáveis e :

- O algoritmo VNS-PPMH apresenta certo grau de sensibilidade às mudanças nas características dos dados. Mais especificamente, este modelo não é afetado pelo número de indivíduos (I) ou pelo número de segmentos (G). No entanto, o modelo é sensível às mudanças quanto à estrutura de categorias. Estruturas com muitas categorias de mesmo tamanho são melhores recuperadas do que aquelas com poucas categorias e de tamanho variado. Nota-se também que o modelo é sensível às distor-

ções adicionadas tanto ao número de pilhas quanto às matrizes de dissimilaridades, mesmo que estas sejam pequenas;

- O algoritmo VNS-PPMHLP é menos sensível aos diferentes fatores aplicados às instâncias. Este não é afetado pelas distorções adicionadas tanto nas pilhas quanto nas distâncias. O algoritmo também se mostra mais robusto na recuperação da estrutura de categorias quando há um grande número de objetos e quando o número total de segmentos G é menor do que 10. Por fim, o VNS-PPMHLP não é afetado em razão do número de indivíduos I . Na verdade, os fatores considerados têm impacto mínimo no valor do ARI, pois a média para este indicador é 0,952. Isto pode ser considerado como referência de perfeita recuperação da informação.

Examina-se, em sequência, a sensibilidade dos modelos aos fatores da simulação, ponderando-se a estimação do ARI calculado em relação às variáveis p :

- O VNS-PPMH apresenta dificuldades em recuperar a pertinência dos indivíduos quando ruídos são adicionados ao número de pilhas, mesmo que estes ruídos sejam pequenos. Também apresenta dificuldades em recuperar estes valores quando o número de categorias presentes nas instâncias é pequeno;
- O VNS-PPMHLP é apenas minimamente mais sensível do que o VNS-PPMH na recuperação dos valores das variáveis p quando o número de segmentos é elevado ($G = 10$) e é realmente impactado quando um alto nível de perturbação é feito no número de pilhas.

Anteriormente, a análise de desempenho do algoritmo mostrou que quando o número de segmentos é elevado ($G = 10$), o algoritmo VNS-PPMHLP tem maior variação em sua função objetivo. Portanto, pode-se supor que esta dificuldade em recuperar os valores das variáveis e e p quando o número de segmentos é elevado mostra que o algoritmo necessita ser melhorado, especialmente para instâncias envolvendo um número maior de segmentos. Uma alternativa, em um primeiro momento, seria executar o VNS-PPMHLP durante mais tempo.

Em linhas gerais, mostrou-se com a análise apresentada que o VNS-PPMHLP é mais robusto que o VNS-PPMH, mesmo considerando suas limitações com relação às instâncias cujo número de segmentos é elevado. Esta piora no desempenho dos algoritmos de *clustering* é frequentemente observada nos casos em que se eleva de forma considerável o número de *clusters*. A próxima seção deste texto apresenta um breve exemplo de aplicação para o PPMHLP.

Instância	I	G	J	Categorias	Perturbação		VNS-PPMHLP		VNS-PPMH	
					Distâncias	Pilhas	ARI		ARI	
							Segmentos	Pilhas	Segmentos	Pilhas
1	150	10	30	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)	0.886	0.903	0.958	0.713
2	300	2	18	All 6	N(0, 0.1)	0	1.000	1.000	1.000	1.000
3	450	2	18	50 % 3, 50 % 6	N(0, 0.05)	0	1.000	1.000	1.000	1.000
4	150	2	18	All 3	N(0, 0.05)	N(0, 0.5)	0.947	0.988	0.947	0.616
5	450	10	18	All 6	N(0, 0.05)	N(0, 1)	0.928	0.911	0.985	0.892
6	150	10	18	50 % 3, 50 % 6	N(0, 0.05)	0	1.000	1.000	1.000	1.000
7	300	2	18	All 6	0	N(0, 0.5)	0.987	0.881	0.987	1.000
8	150	10	18	50 % 3, 50 % 6	0	N(0, 1)	0.166	0.979	0.167	0.182
9	300	10	30	All 3	N(0, 0.05)	N(0, 0.5)	0.700	0.787	0.852	0.727
10	450	6	18	All 3	N(0, 0.1)	N(0, 1)	0.771	0.844	0.783	0.267
11	150	6	30	All 6	N(0, 0.1)	0	1.000	1.000	1.000	1.000
12	300	10	18	All 3	0	0	1.000	1.000	1.000	1.000
13	450	10	18	All 6	N(0, 0.1)	0	1.000	1.000	1.000	1.000
14	300	6	18	50 % 3, 50 % 6	0	N(0, 1)	0.960	0.985	0.878	0.657
15	300	2	30	All 6	N(0, 0.05)	N(0, 1)	0.934	0.985	0.871	0.716
16	450	2	30	50 % 3, 50 % 6	0	N(0, 1)	0.879	0.969	0.293	0.846
17	300	6	18	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)	0.900	0.958	0.711	0.818
18	300	6	30	50 % 3, 50 % 6	N(0, 0.05)	0	1.000	1.000	1.000	1.000
19	150	6	18	All 6	0	N(0, 0.5)	0.968	0.987	0.947	1.000
20	450	6	30	All 3	0	0	1.000	1.000	1.000	1.000
21	150	2	30	All 3	N(0, 0.1)	N(0, 1)	0.821	0.955	0.797	0.200
22	450	2	18	50 % 3, 50 % 6	N(0, 0.1)	N(0, 0.5)	0.956	0.990	0.247	1.000
23	450	6	18	All 3	N(0, 0.05)	N(0, 0.5)	0.783	0.882	0.860	0.668
24	300	10	18	All 3	N(0, 0.1)	N(0, 1)	0.751	0.862	0.879	0.417
25	150	6	18	All 6	N(0, 0.05)	N(0, 1)	0.890	0.913	0.868	0.897
26	150	2	18	All 3	0	0	1.000	1.000	1.000	1.000
27	450	10	30	All 6	0	N(0, 0.5)	0.889	0.926	0.946	1.000

Tabela 3.4: Simulação de Monte Carlo: precisão dos algoritmos

Fator		VNS-PPMHLP				VNS-PPMH			
		Beta	valor- <i>t</i>	valor- <i>p</i>		Beta	valor- <i>t</i>	valor- <i>p</i>	
<i>Intercepto</i>		1.084	37.521	.000	*	1.081	37.826	.000	*
<i>Indivíduos</i>	300	-.030	-1.426	.174		-.023	-1.134	.275	
(padrão: 150)	450	-.023	-1.085	.295		.000	.005	.996	
<i>Segmentos</i>	6	-.022	-1.067	.303		-.028	-1.378	.188	
(padrão: 2)	10	-.045	-2.136	.050	*	-.008	-.387	.704	
<i>Número de Objetos</i>	30	-.007	-.404	.692		.001	.051	.960	
(padrão: 18)									
<i>Número de Categorias</i>	Todas 3	-.052	-2.479	.026	*	-.091	-4.383	.001	*
(padrão: 50% 3, 50% 6)	Todas 6	-.020	-.964	.350		.041	1.971	.067	*
<i>Perturbações (Distâncias)</i>	Pequeno	-.029	-1.397	.183		-.054	-2.598	.020	*
(padrão: sem perturbação)	Grande	-.024	-1.143	.271		-.105	-5.086	.000	*
<i>Perturbações (Pilhas)</i>	Pequeno	-.078	-3.718	.002	*	-.078	-3.800	.002	*
(padrão: sem perturbação)	Grande	-.067	-3.190	.006	*	-.159	-7.709	.000	*
R^2	0.678					0.893			
R^2 Ajustado	0.448					0.882			
ARI Médio	0.952					0.913			
ARI desvio padrão	0.059					0.104			

Tabela 3.5: Simulação de Monte Carlo: fatores que influenciam na recuperação da estrutura de categorias (e_{jk}^g)

Fator		VNS-PPMHL P			VNS-PPMH		
		Beta	valor- <i>t</i>	valor- <i>p</i>	Beta	valor- <i>t</i>	valor- <i>p</i>
<i>Intercepto</i>		.956	9.216	.000 *	.985	9.278	.000 *
<i>Indivíduos</i>	300	.062	.822	.424	.079	1.031	.319
(padrão: 150)	450	.059	.783	.446	.111	1.450	.168
<i>Segmentos</i>	6	-.028	-.374	.714	.009	0.122	.905
(padrão: 2)	10	-.134	-1.784	.095 *	-.031	-0.405	.691
<i>Número de Objetos</i>	30	.012	.181	.858	.018	0.278	.785
(padrão: 18)							
<i>Número de Categorias</i>	Todos 3	.003	.038	.970	-.142	-1.855	.083 *
(padrão: 50% 3, 50% 6)	Todos 6	.094	1.259	.227	.116	1.507	.152
<i>Perturbações (Distâncias)</i>	Pequena	.037	.492	.630	-.018	-0.239	.814
(padrão: sem perturbação)	Grande	.026	.348	.733	-.095	-1.245	.232
<i>Perturbações (Pilhas)</i>	Pequena	-.109	-1.460	.165	-.134	-1.747	.101 *
(padrão: sem perturbação)	Grande	-.211	-2.816	.013 *	-.428	-5.589	.000 *
<i>R²</i>	0.494				0.763		
<i>R² Ajustado</i>	0.124				0.590		
<i>ARI médio</i>	0.893				0.813		
<i>ARI desvio padrão</i>	0.170				0.254		

Tabela 3.6: Simulação de Monte Carlo: fatores que influenciam na recuperação da pertinência aos segmentos (p^{ig})

3.4.4 Exemplo de aplicação

Para ilustrar a aplicação do PPMHLP e VNS-PPMHLP, considera-se uma tarefa de triagem realizada a partir de alguns dos produtos vendidos pela empresa *Vital Vittles*, inaugurada em 1973. Esta empresa foi uma das primeiras no segmento de loja de conveniências a atender os estudantes da Georgetown University, nos Estados Unidos. A empresa vende comida congelada, lanches para levar, salgadinhos e doces, além de suprimentos diversos para o lar.

Um dos gêneros de produtos que a empresa mais vende são chocolates, especialmente as seguintes opções:

1. Almond Joy;
2. Baby Ruth;
3. Butterfinger;
4. Hershey (Almond);
5. Hershey (Plain);
6. Junior Mints;
7. Kit Kat;
8. M & M (Peanut);
9. M & M (Plain);
10. Mars Bar;
11. Milky Way;
12. Mounds Bar;
13. Crunch (Nestlé);
14. Oh Henry!;
15. Payday;
16. Reece's Cups;
17. Snickers;
18. Three musketeers;
19. Twix, e;
20. York Mint.

No contexto do gerenciamento da empresa, observa-se que regularmente as opções oferecidas mudam, isto é, algumas marcas de chocolate podem deixar de ser vendidas ou novas marcas podem ser incluídas dentre as opções. Organizar estes produtos em uma prateleira, ou mesmo definir qual o melhor arranjo para disposição em uma máquina de auto-venda, são decisões que podem influenciar na lucratividade das operações da empresa.

Considerando-se, portanto, a necessidade de estabelecer uma disposição otimizada dos produtos em gôndolas, aplicou-se os algoritmos VNS-PPMHLP e VNS-PPMH a dados obtidos a partir de uma tarefa de triagem realizada para estes produtos. Esta tarefa envolveu 189 estudantes da referida universidade e as 20 opções de produtos apresentadas. Os resultados dos experimentos são apresentados nas próximas seções deste texto.

Seleção de um modelo

A primeira decisão a ser tomada quando da aplicação dos modelos PPMHLP e PPMH é escolher qual a melhor configuração a respeito do número de segmentos a ser utilizado, isto é, escolher o valor do parâmetro G . Para tal, considerou-se executar estes algoritmos para os dados obtidos com valores de G variando de 1 a 8. Para cada um destes valores, 10 execuções de ambos os algoritmos foram realizadas. O tempo limite para os algoritmos foi definido em 10 minutos.

A Tabela 3.7 mostra os custos obtidos a partir de cada uma destas execuções. São apresentados também nesta tabela o melhor e o pior custo para cada valor de G , bem como a distância em porcentagem entre estes. Pode-se observar que ambos os algoritmos apresentam variações ao se comparar o custo da pior e melhor solução em relação a cada valor de G . No entanto, um teste- t para dados pareados (em relação a G) mostra que a média destas variações entre a pior e melhor solução do VNS-PPMHLP ($\bar{M}_{VNS-PPMHLP} = 0,17\%$) é menor que a média destas distâncias observadas para o algoritmo VNS-PPMH ($\bar{M}_{VNS-PPMH} = 0,72\%$), pois valor- $p = 0,008$.

Diferentemente do resultado observado para os dados simulados, o VNS-PPMH mostrou-se menos estável quando o número de segmentos G é aumentado. O algoritmo VNS-PPMHLP, no entanto, demonstrou maior estabilidade. Mostrou-se anteriormente que o algoritmo era pouco sensível à variação no número de segmentos, respeitando-se a condição $G < 10$.

Em termos práticos, no entanto, nem sempre é necessária a resolução do problema para um número elevado de segmentos, pois pode-se adotar um critério para a escolha do número de partições, assim como feito por Blanchard et al. (2012): a observação do comportamento da função objetivo à medida em que o valor do parâmetro G é aumentado. Um acréscimo em G tende a reduzir o custo da função objetivo. Desta forma, busca-se o ponto (valor de G) em que há a maior queda em relação ao ponto anterior ($G - 1$). Testes computacionais, utilizando-se um algoritmo de força bruta, sugerem que sempre haverá queda ou manutenção da função objetivo à medida em que o número de segmentos é acrescido.

G	Execuções VNS-PPMHLP										Pior	Melhor	Dist.
	1	2	3	4	5	6	7	8	9	10	Custo	Custo	%
1	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	2493.59337	0.00%
2	2317.91752	2317.91752	2318.233	2318.233	2318.358	2318.358	2318.358	2318.358	2318.358	2318.358	2318.358	2317.91752	0.02%
3	2300.34736	2300.34736	2300.55115	2300.55115	2300.55863	2300.55863	2303.3667	2303.3667	2303.47171	2303.47171	2303.47171	2300.34736	0.14%
4	2283.15735	2283.15735	2283.86808	2283.86808	2284.44932	2284.44932	2288.46472	2288.46472	2290.22425	2290.22425	2290.22425	2283.15735	0.31%
5	2276.36759	2276.36759	2276.6772	2276.6772	2277.09591	2277.09591	2278.28891	2278.28891	2278.44872	2278.44872	2278.44872	2276.36759	0.09%
6	2266.52561	2266.52561	2272.76394	2272.76394	2273.07408	2273.07408	2273.07408	2273.44529	2273.98825	2273.98825	2273.98825	2266.52561	0.33%
7	2262.31673	2262.31673	2264.29185	2265.07753	2265.11295	2265.11295	2265.55577	2265.55577	2266.86921	2266.86921	2266.86921	2262.31673	0.20%
8	2256.50771	2256.50771	2259.17076	2259.17076	2260.50398	2260.50398	2261.62318	2261.62318	2261.9713	2261.9713	2261.9713	2256.50771	0.24%
G	Execuções VNS-PPMH										Pior	Melhor	Dist.
	1	2	3	4	5	6	7	8	9	10	Custo	Custo	%
1	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	2298.36	0.00%
2	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	2219.37	0.00%
3	2249.63	2249.63	2249.63	2249.63	2220.69	2249.63	2220.69	2249.63	2249.63	2249.63	2249.63	2220.69	1.29%
4	2240.43	2239.51	2241.64	2240.43	2240.43	2239.51	2240.43	2240.43	2244.69	2240.43	2244.69	2239.51	0.23%
5	2239.68	2231.69	2221.15	2245.18	2219.14	2245.18	2239.91	2237.49	2239.91	2233.2	2245.18	2219.14	1.16%
6	2219.51	2232.37	2215.73	2225.54	2217.62	2232.79	2223.57	2228.92	2221.4	2226.59	2232.79	2215.73	0.76%
7	2210.84	2204.33	2215.42	2201.95	2222.89	2214.36	2222.35	2231.44	2223.89	2232.35	2232.35	2201.95	1.36%
8	2202.1	2212.56	2210.85	2211.22	2200.01	2220.18	2220.55	2208.72	2206.68	2208.77	2220.55	2200.01	0.92%

Tabela 3.7: Exemplo de aplicação: 10 execuções para o VNS-PPMHLP e VNS-PPMH

Considerando-se a regra descrita, a Tabela 3.8 mostra o custo da melhor solução obtida para cada um dos valores de G para ambos os algoritmos. Apresenta-se também o percentual de melhora observado na função objetivo em relação ao valor de G e seu ponto anterior ($G - 1$). Ponderando-se tais resultados, vê-se que ambos os modelos sugerem $G = 2$ como o número ideal de segmentos, dado que acréscimos a este valor não geram melhoras significativas no custo da solução. No caso do VNS-PPMHLP, a escolha de $G = 2$ é um tanto evidente, pois para $G > 2$, as melhorias observadas na função objetivo não ultrapassam 1%.

Adicionalmente, para as soluções de ambos os modelos com $G = 2$, calculou-se o ARI com base nas variáveis e . Para estes valores, o VNS-PPMHLP mostra-se um pouco superior, pois $ARI_{VNS-PPMHLP} = 0,3143$ e $ARI_{VNS-PPMH} = 0,2931$. Isto representa uma melhora de 7,23% em termos de precisão ao recuperar a estrutura de categorias estabelecida entre os objetos durante a tarefa de triagem. Esta diferença pode ser explicada em razão de os algoritmos apresentarem diferenças quanto às regras para a definição do número de categorias em cada segmento, o que influencia a alocação dos indivíduos. O ARI em relação às variáveis p , que representam a segmentação dos indivíduos, diferentemente, não pode ser calculado para estes dados reais, uma vez que não se dispõe de valores de referência.

Deve-se lembrar que a definição do parâmetro δ (fator de penalidade) pode afetar positiva ou negativamente a forma como as informações são recuperadas no VNS-PPMH. Isto é, diferentes valores de δ geram diferentes informações. O cálculo do ARI entre os valores das variáveis p recuperados pelo VNS-PPMHLP e VNS-PPMH sugere considerável discordância em relação à segmentação dos consumidores obtida para cada um destes, uma vez que $ARI(p_{VNS-PPMHLP}, p_{VNS-PPMH}) = 0,3413$.

Destaca-se que Blanchard et al. (2012) propuseram duas maneiras para determinar o valor de δ . Uma delas é utilizar o valor médio de todas as dissimilaridades d_{jk}^i . A outra estratégia sugerida pelos autores envolve uma etapa de pré-processamento baseada em um algoritmo livre de derivada, que testa variações em δ enquanto maximiza um critério externo. Na prática, os autores observaram a partir de seus experimentos que ambos os métodos geram resultados semelhantes. Logo, sugeriu-se que a estratégia mais simples, ou seja, utilizar a média das dissimilaridades, era mais satisfatória. Para estes experimentos, portanto, esta foi a estratégia aplicada.

A próxima seção deste texto apresenta uma descrição da estrutura de categorias recuperada por cada um dos algoritmos, assim como alguns detalhes sobre a segmentação dos consumidores observada.

G	PFVNS	% Melhora	VNS-PPMH	% Melhora
1	2493.59	-	2929.25	-
2	2317.92	7.05%	2542.14	13.22%
3	2300.35	0.76%	2417.27	4.91%
4	2283.16	0.75%	2365.69	2.13%
5	2276.37	0.30%	2333.86	1.35%
6	2266.53	0.43%	2316.06	0.76%
7	2262.32	0.19%	2306.09	0.43%
8	2256.51	0.26%	2292.62	0.58%

Tabela 3.8: Exemplo de aplicação: seleção de modelo (G)

Uma possível interpretação dos resultados

A estrutura de categorias obtida a partir do VNS-PPMHLP é apresentada na Tabela 3.9. Esta solução é composta de dois segmentos: *Novatos* (Segmento 1, com $N = 69$ indivíduos) e *Especialistas* (Segmento 2, com $N = 120$). Esta classificação e nominação de grupos se dá em razão de pesquisa efetuada com os estudantes após a realização do processo de categorização. Os indivíduos deveriam qualificar as seguintes Sentenças:

1. **Sentença 1:** Estou confiante em relação ao meu julgamento dos produtos.
2. **Sentença 2:** Gosto mais de comidas salgadas que doces.
3. **Sentença 3:** Conheço mais acerca destes produtos que meus colegas.

As respostas deveriam ser dadas utilizando-se uma escala, de 1 a 5, em que 1 representa "discordo fortemente" e 5 representa "concordo fortemente". Para os Novatos, o valor médio do julgamento em relação à Sentença 1 foi $\bar{M}_{Novatos} = 4.55$. Para os Especialistas este valor foi de $\bar{M}_{Especialistas} = 4.92$. A diferença entre estes dois valores é significativa, com valor- $t(184) = 1.72$ e valor- $p = 0.09$. Para a Sentença 2 $\bar{M}_{Novatos} = 1.79$, $\bar{M}_{Especialistas} = 1.42$, com valor- $t(184) = 1.68$ e valor- $p = 0.09$, isto é, os Novatos são menos adeptos aos doces que os Especialistas. Por fim, para a Sentença 3, $\bar{M}_{Novatos} = 2.94$ e $\bar{M}_{Especialistas} = 3.19$, com valor- $t(184) = 1.72$ e valor- $p = 0.09$.

Obviamente, a denominação dada a cada segmento é uma decisão subjetiva, e cada decisor pode analisar os dados obtidos de forma diferente. Como fecho a esta Seção, algumas observações acerca de cada estrutura de categorias obtida por meio dos algoritmos são apresentadas.

Segmento 1: Novatos

Este segmento apresenta 5 categorias. Como mostrado, estes indivíduos são consumidores menos assíduos de chocolates. Pode-se observar que a classificação por eles feita se baseia fortemente em três aspectos: (a) ingrediente principal (quando este ingrediente

é bem evidente), (b) estrutura física e (c) popularidade. Por exemplo, a Categoria 1 engloba produtos mentolados: Junior Mints e York Mint, sendo o primeiro a mediana. O agrupamento destes itens parece ser bem óbvia, dado seus nomes. A Categoria 2 inclui chocolates em barras, tanto pequenas quanto grandes: Hershey plain (mediana), Kit Kat, Hershey (Almond) e Nestle Crunch. A Categoria 3 inclui chocolates pequenos, não em barras, como M&M Peanuts (mediana) M&M Regular e Reese's Cups.

As categorias 4 e 5 representam doces com cobertura, no entanto diferem em relação à popularidade das marcas. A categoria 4 apresenta os chocolates mais populares, visto que os indivíduos declararam que conheciam todos estes itens. Pode-se ver que o chocolate Snickers é a mediana desta categoria. Este produto é, de fato, o chocolate mais vendido no mundo. Os outros elementos desta categoria incluem Butterfinger, Milky Way, 3 Musketeers e Twix. Por fim, a categoria 5 engloba os chocolates com menor popularidade: Mounds (mediana), Almond Joy, Baby Ruth, Mars, Oh Henry! e Payday.

Uma pesquisa realizada com os participantes da tarefa de triagem sugere que os chocolates da Categoria 4, mais populares, têm 16,23% de chance de serem consumidos ao menos uma vez por mês. Os chocolates classificados na Categoria 5, contrariamente, têm apenas 3%. Ainda, em média, 28% dos chocolates desta categoria são desconhecidos aos indivíduos.

Segmento 2: Especialistas

Assim como os Novatos, os membros deste segmento formaram uma categoria com chocolates mentolados, incluindo Junior Mints e York Mint. No entanto, sua classificação como um todo é mais complexa, pois apresenta duas categorias adicionais. Para estes indivíduos, a Categoria 2 é composta por chocolates nogados: Milky Way (mediana), Mars, Snickers e Three Musketeers. A Categoria 3 contém chocolates com base em amêndoas e côco: Almond Joy (mediana), Hershey (Almond) e Mounds Bar.

A Categoria 4 inclui chocolates crocantes, como Kit Kat (mediana), Nestle's Crunch e Twix, sendo que estes chocolates são conhecidos por esta característica. A Categoria 5 contém os chocolates Reese's cups (mediana) e butterfinger, dois chocolates conhecidos por serem feitos com manteiga de amendoim. A categoria 6 possui os chocolates Payday (mediana), Baby Ruth e Oh Henry, todos caracterizados pela mistura de caramelo e amendoim. Por fim, a Categoria 7 contém os chocolates de pequeno porte, como M&M, Hershey Plain e M&M Peanuts.

Implicações práticas

A breve análise apresentada acerca de cada estrutura de categorias sugere que a divisão dos indivíduos em dois grupos, os *Novatos* e os *Especialistas* pode estar correta, pois os *Novatos* não demonstraram um conhecimento mais amplo em relação às características dos doces, como por exemplo o sabor e a composição. A presença deste conhecimento é facilmente percebida no segmento dos *Especialistas*. Os *Novatos* basicamente classificaram os doces pela popularidade, nome (caso dos chocolates mentolados) e forma física.

Além de observações deste tipo, os resultados gerados pelo VNS-PPMHL podem auxiliar o tomador de decisão a responder outras perguntas, como por exemplo:

- Meu produto é o líder em seu segmento de mercado?
- Quais são meus concorrentes?
- Qual o perfil dos consumidores e qual sua percepção acerca dos produtos?

Dentre estas três perguntas, a terceira tem relação direta com o propósito deste trabalho e ao trabalho de Blanchard et al. (2012). Se a opinião dos indivíduos fosse tratada sem considerar a heterogeneidade, isto é, agregando-se as matrizes de dissimilaridades, seria inviável estabelecer a estrutura de categorias e relacioná-la a cada grupo de indivíduos. Para o exemplo dado, mostra-se claramente que há divergências quanto à forma de percepção dos produtos e suas relações. Ambas as informações podem ser úteis em um cenário real, no qual o tomador de decisões poderia criar novas estratégias para maximizar suas vendas.

Um exemplo disto seria o reposicionamento dos produtos em diferentes lugares. Para os *Novatos*, que consomem chocolates eventualmente, os produtos podem ser colocados no caixa da loja, pois na maioria das vezes estes indivíduos não vão ao estabelecimento com o intuito de comprar tal produto. Como o espaço no caixa é geralmente reduzido, aqueles produtos que se mostraram mais representativos em cada uma das categorias podem ser considerados. Ainda, como estes são os produtos que melhor representam um segmento, pode-se pensar que estes não deverão ficar indisponíveis por falta de estoque.

Para os *Especialistas*, pode-se ter uma seção dentro da loja onde todos os chocolates serão dispostos, a fim de que o consumidor com este perfil possa ir buscar e encontrar com mais facilidade aquilo que procura. Os *Especialistas*, como mostrado, são aqueles que consomem chocolates com mais frequência e sabem exatamente qual a composição dos produtos, logo saberão escolher com maior facilidade o que desejam. Se a prateleira estiver bem disposta, ajudará o cliente a encontrar o produto desejado.

Segmento 1 - Novatos

Nome da Categoria	Mediana	Membros				
Mentolados	Junior Mints	York Mint				
Tabletes	Hershey (Plain)	Kit Kat	Hershey (Almond)	Nestle's Crunch		
Doces Pequenos	M & M (Peanut)	M & M (Plain)	Reese's Cups			
Populares	Snickers	Butterfinger	Milky Way	Three Musketeers	Twix	
Pouco Populares	Mounds Bar	Almond Joy	Baby Ruth	Mars	Oh Henry!	Payday

Segmento 2 - Especialistas

Nome da Categoria	Mediana	Membros		
Mentolados	Junior Mints	York Mint		
Nogados	Milky Way	Mars	Snickers	Three Musketeers
Amêndoas/Côco	Almond Joy	Hershey (Almond)	Mounds Bar	
Crocantes	Kit Kat	Nestle’s Crunch	Twix	
Manteiga de Amendoim	Reese’s Cups	Butterfinger		
Amendoim e Caramelo	Payday	Baby Ruth	Oh Henry!	
Doces Pequenos	M & M (Plain)	Hershey (Plain)	M & M (Peanut)	

Tabela 3.9: Exemplo de aplicação: estrutura de categorias

Capítulo 4

Uma extensão do problema das p -medianas heterogêneo livre de penalidade a ambientes *fuzzy*

Neste Capítulo, sugere-se uma extensão do PPMHLP a ambientes *fuzzy*, buscando-se construir um modelo em que novos aspectos em relação à recuperação das estruturas de categorias e segmentação dos indivíduos possam ser analisados pelo tomador de decisão. A motivação maior para a proposição deste novo modelo é dada pelo fato de que indivíduos alocados a diferentes segmentos podem ter julgamentos similares em relação a algumas das categorias estabelecidas entre os objetos.

Uma boa demonstração disto é o caso mostrado no exemplo de aplicação apresentado no Capítulo 3, no qual os dois segmentos identificados possuem uma categoria em comum em relação aos chocolates mentolados. Para o caso em que este resultado foi analisado, coube ao responsável pela análise perceber tal fato. Em algumas situações, no entanto, poderá não ser tão trivial observar esta ocorrência.

A apresentação do PPMHLP como um problema de *clustering fuzzy* (PCF), portanto, poderá ser considerada como uma implementação complementar. Esta abordagem é baseada na teoria dos conjuntos *fuzzy*, também conhecida como lógica nebulosa, a qual permite tratar a imprecisão dos dados e é bastante consolidada na literatura em relação à sua utilização em problemas de *clustering*, sendo que a primeira publicação relacionada a estes métodos data da década de 70, em [Dunn 1973].

Destaca-se, no entanto, que as proposições aqui feitas são consideradas como direções futuras a esta pesquisa, logo o novo modelo sugerido poderá sofrer grandes modificações com a continuidade deste trabalho. O objetivo principal deste Capítulo é apresentar novas ideias em relação ao PPMHLP, sugerindo-se que a incorporação de um grau de imprecisão ao modelo poderá ser de grande valia para a análise a ser feita pelo tomador de decisões.

4.1 Motivação

Neste trabalho, o problema das p -medianas heterogêneo proposto por Blanchard et al. (2012) foi remodelado. Tanto este novo modelo apresentado, quanto o modelo original, baseiam-se no trato de dados oriundos de uma tarefa de triagem, para a qual um grupo de I indivíduos deve categorizar J objetos. Nesta técnica, cada indivíduo pode estabelecer entre os objetos quantas categorias achar conveniente. Como saída, obtém-se neste processo matrizes de distâncias, as quais apresentam o grau de similaridade par-a-par entre os objetos. Estas matrizes, por sua vez, são individuais. Logo, haverá uma matriz relacionada ao julgamento de cada indivíduo sobre os objetos.

O grande diferencial do PPMH e do PPMHLP em relação aos modelos de *clustering* tradicionais é que ambos eliminam a necessidade de agregação destas matrizes, provendo suporte ao trato da heterogeneidade relacionada à opinião dos indivíduos. Isto permitiu identificar diferentes estruturas de categorias entre os diversos objetos e relacionar estas diferentes estruturas a diferentes grupos de indivíduos. Pode-se, portanto, analisar o perfil destes, bem como a forma como tais indivíduos percebem a relação entre os vários objetos. Porém, esta informação dada pelos novos modelos, embora seja de grande utilidade, como demonstrado no exemplo de aplicação dado (ver Capítulo 3), pode mascarar algumas peculiaridades em relação à informação contida nos dados.

Para demonstrar isto, duas linhas de pensamento serão analisadas a seguir. A primeira delas está relacionada ao fato de que os indivíduos envolvidos no processo de julgamento de similaridades entre os objetos podem basear sua decisão em diferentes fatores. Produzirão, portanto, julgamentos distintos, o que neste trabalho é tratado como "heterogeneidade". Esta proposição foi sustentada por Blanchard et al. (2012), relacionando, por exemplo, o nível de conhecimento, [Sujan & Dekleva 1987], a idade [John & Sujan 1990], diferentes situações [Medin & Schaffer 1978, Ross & Murphy 1999], o humor [Isen 2012], entre outros fatores, a diferentes julgamentos.

A segunda, é que ao assumir que as variáveis p dos modelos PPMHLP e PPMH são binárias, fica implícito que se um indivíduo i_1 está alocado em um segmento diferente de outro indivíduo i_2 , estes apresentam opiniões heterogêneas, ou seja, divergentes. No entanto, sugere-se aqui que pode haver certo grau de concordância na opinião destes dois indivíduos.

Para ilustrar tal situação, considera-se o seguinte cenário fictício para a aplicação do PPMHLP: um grupo de $I = 3$ indivíduos participou da aplicação de uma tarefa de triagem, na qual quatro restaurantes ($J = 4$) deveriam ser categorizados. Os 4 restaurantes são descritos brevemente nos itens que seguem:

- **Restaurante China5** ($j = 1$): este restaurante vende apenas pratos chineses tradicionais;
- **Restaurante Food&Fast** ($j = 2$): o restaurante Food&Fast é especializado em servir pratos executivos, sendo uma boa opção para o almoço. Este restaurante se destaca por, além de vender refeições, vender *fast food*, como hambúrgueres e cachorros-quentes;
- **Restaurante TopMex** ($j = 3$): o TopMex vende apenas comida mexicana, unicamente no formato *fast food*;
- **Restaurante 100% Brasil** ($j = 4$): este restaurante oferece apenas bufê ao meio-dia e à noite, isto é, refeições em sua forma mais tradicional.

Como resultado do processo de julgamento em relação aos objetos, supõe-se que os quatro indivíduos tenham construído suas pilhas de acordo com o que é apresentado na Tabela 4.1. Adicionalmente, considera-se que também tenham dado nomes específicos às suas pilhas, definindo o que cada uma das categorias representa segundo sua percepção. Para estes dados apresentados, executou-se o modelo PPMHLP por meio do *solver* CPLEX 12.5, o qual retornou a solução ótima. Destaca-se que a solução ótima foi obtida em razão de o tamanho da instância ser muito pequeno. A estrutura de categorias e a segmentação dos consumidores gerados pelo modelo é apresentada na Tabela 4.2.

Indivíduo $i = 1$	Definição	Elementos	
	Comidas típicas	China5	
	Fast food	Food&Fast	Top Mex
	Bufês	100% Brasil	
Indivíduo $i = 2$	Definição	Elementos	
	Comidas típicas	China5	TopMex
	Refeições	Food&Fast	100% Brasil
Indivíduo $i = 3$	Definição	Elementos	
	Comidas típicas	China5	
	Refeições	Food&Fast	100% Brasil
	Fast food	TopMex	

Tabela 4.1: Estruturas de categorias formada pelos indivíduos 1, 2 e 3

Ao definir que os indivíduos 1 e 3 estão alocados ao segmento 1 e que o indivíduo 2 está alocado ao segmento 2, pode-se considerar que os indivíduos 1 e 3 possuem uma opinião mais similar em relação às suas categorias, diferindo do modo de pensar do indivíduo 2. No entanto, ao verificar as pilhas feitas por estes três indivíduos durante a tarefa

de classificação, pode-se ver que todos concordam que o China5 é um restaurante de comidas típicas. Além disto, os três indivíduos também concordam que o restaurante 100% Brasil pertence à categoria de refeições tradicionais, como almoço ou janta.

Neste caso, caberá ao decisor analisar manualmente os dados e ponderar que há uma inconsistência, embora parcial, na segmentação dos indivíduos. Estes não divergem totalmente em suas estruturas de categorias. O fato a ser notado é que para o PPMHLP existe uma fronteira sólida separando os indivíduos apesar destes compartilharem opiniões. A partir da próxima seção deste texto, formula-se o PPHMLP-*fuzzy* (Problema das p -Medianas Heterogêneo Livre de Penalidade *Fuzzy*), que permitirá visualizar uma classificação dos indivíduos considerando opiniões parcialmente consensuais.

Segmento 1	Membros:	indivíduos 1 e 3	
	Descrição	Elementos	
Categoria 1	Comidas Típicas	<i>China5</i>	
Categoria 2	Fast Food	<i>TopMex</i>	Food&Fast
Categoria 3	Refeições	<i>100% Brasil</i>	
Segmento 2	Membros:	indivíduo 2	
	Descrição	Elementos	
Categoria 1	Comidas Típicas	<i>China5</i>	TopMex
Categoria 2	Refeições	<i>100% Brasil</i>	Food&Fast

Tabela 4.2: Resultado do PPMHLP para o exemplo dado

4.2 Um modelo *fuzzy* para o PPMHLP

No Capítulo 1 deste texto, propôs-se a modificação do PPMH de Blanchard et al. (2012) por meio da eliminação do segundo termo em sua função objetivo, dada em (1.8). Este termo tinha como utilidade otimizar o número de medianas a ser recuperado pelo modelo em cada um dos segmentos g . A partir desta eliminação, introduziu-se uma nova restrição, dada por (2.2), que define que o número máximo de medianas para cada segmento g deve ser menor ou igual ao piso do número médio de pilhas feitas pelos indivíduos alocados a este segmento.

Estas manipulações matemáticas, por sua vez, geraram o PPMHLP, dado por:

$$\text{Minimize } M = \sum_{i=1}^I \sum_{g=1}^G p^{ig} \left[\sum_{j=1}^J \sum_{k=1}^J d_{jk}^i e_{jk}^g \right], \quad (4.1)$$

sujeito a

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad \forall j = 1, \dots, J, \quad (4.2)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (4.3)$$

$$\sum_{j=1}^J e_{jj}^g \leq \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}}, \quad \forall g = 1, \dots, G, \quad (4.4)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad \forall j = 1, \dots, J, \quad \forall k = 1, \dots, J, \quad (4.5)$$

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad (4.6)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad \forall j = 1, \dots, J, \quad \forall k = 1, \dots, J, \quad (4.7)$$

$$p^{ig} \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad \forall i = 1, \dots, I. \quad (4.8)$$

Para o modelo mostrado, sugere-se sua transformação para a forma de um PCF por meio da introdução de um expoente m às variáveis p^{ig} em sua função objetivo, elemento clássico de um PCF. Desta forma, os valores das variáveis p , que deverão ser contínuas no intervalo $[0, 1]$, representarão o grau de pertinência de cada um dos I indivíduos em relação aos G segmentos. Isto permitirá a atribuição de um indivíduo a um ou mais segmentos. Adicionando-se esta modificação na função objetivo do PPMHLP, obtém-se:

$$\text{Minimize } M = \sum_{i=1}^I \sum_{g=1}^G (p^{ig})^m \left[\sum_{j=1}^J \sum_{k=1}^J d_{jk}^i e_{jk}^g \right], \quad (4.9)$$

em que m representa o grau de imprecisão (*fuzzyness*) que será considerado para a segmentação dos indivíduos. Quanto maior este valor, maior este grau de imprecisão. A definição do valor deste parâmetro está fortemente ligada ao grau de separação entre os diversos segmentos e objetos.

Sherali & Desai (2005) exemplificam a definição de m : caso os pontos ou objetos estejam claramente divididos em vários *clusters*, pode-se considerar que o problema associado será um problema de *clustering* tradicional, com $m = 1$, em que o grau de pertinência de um indivíduo em um segmento deverá ser ou 0 ou 1. No entanto, se o nível de sobreposição entre os objetos ou pontos for elevado, um valor maior para m poderá ser necessário para que se possa extrair uma informação relevante para dado problema. No

entanto, quando $m \rightarrow \infty$, o valor de pertinência de cada objeto em cada um dos G segmentos a serem considerados tenderá ao valor $1/G$, valor este que inutilizará o modelo [Höppner et al. 1999].

Para esta primeira versão do modelo PPMHLP-*fuzzy*, será considerado $m = 2$. Esta definição é baseada no fato de que este valor é amplamente adotado na literatura. Adicionalmente, como este é um estudo preliminar, objetiva-se verificar a que resultados o valor $m = 2$ poderá levar. Com a continuidade desta pesquisa, uma estratégia mais apurada para definir o valor de m será buscada, podendo-se avaliar com maior clareza se esta escolha é apropriada.

Como mencionado anteriormente, a modificação realizada na função objetivo em (4.9), requer a substituição da restrição de integralidade dada em (4.8) por

$$p^{ig} \in [0, 1], \quad \forall g = 1, \dots, G; \quad \forall i = 1, \dots, I \quad (4.10)$$

para que se possa obter um valor fracionário para as variáveis p^{ig} . Este valor fracionário representará, no novo modelo, o grau de pertinência do indivíduo i em um segmento g . Deve-se notar que apenas a atribuição dos indivíduos a mais de um segmento será considerada, e não a atribuição de objetos a mais de uma categoria, pois as variáveis e permanecerão como binárias.

No entanto, como os indivíduos poderão ser parcialmente atribuídos a mais de um segmento, será possível observar a designação de objetos em mais de uma estrutura de categorias, as quais serão obviamente relacionadas aos indivíduos. A partir das modificações sugeridas em (4.9) e (4.10), o modelo *fuzzy* para o PPMHLP, para um valor m qualquer, é dado por

$$\text{Minimize } M = \sum_{i=1}^I \sum_{g=1}^G (p^{ig})^m \left[\sum_{j=1}^J \sum_{k=1}^J d_{jk}^i e_{jk}^g \right], \quad (4.11)$$

sujeito a

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (4.12)$$

$$\sum_{g=1}^G p^{ig} = 1, \quad \forall i = 1, \dots, I, \quad (4.13)$$

$$\sum_{j=1}^J e_{jj}^g \leq \frac{\sum_{i=1}^I c^i p^{ig}}{\sum_{i=1}^I p^{ig}}, \quad \forall g = 1, \dots, G, \quad (4.14)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad k = 1, \dots, J, \quad (4.15)$$

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad (4.16)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad k = 1, \dots, J, \quad (4.17)$$

$$p^{ig} \in [0, 1], \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad (4.18)$$

em que (4.11) minimiza a soma das dissimilaridades relacionadas a associar um objeto j a outro objeto k em um segmento g , condicionado à pertinência total ou parcial do indivíduo i em cada um destes segmentos. As restrições dadas em (4.12) impõem que todo o objeto j deverá estar associado obrigatoriamente a um objeto k em cada um dos segmentos g . As restrições dadas em (4.13) impõem que todos os indivíduos i deverão ser designados a um ou mais segmentos g .

Diferentemente da restrição (4.4) no modelo do PPMHLP, a restrição dada por (4.14) garante que o número de medianas para cada um dos segmentos g será menor ou igual ao piso da média ponderada em relação ao número de pilhas feitas pelos indivíduos alocados a este segmento durante a tarefa de triagem. Considera-se, neste caso, que o valor das variáveis p^{ig} serão os pesos. Logo, se um indivíduo i for atribuído a um segmento g com um grau de pertinência elevado, isto é, próximo de 1, o número de pilhas c^i deste indivíduo terá maior peso na determinação do número total de medianas permitidos para o segmento g .

A desigualdade em (4.15) garante que um objeto j só será atribuído a um objeto k em um segmento g , se este objeto k for uma das medianas dentro deste segmento. A restrição dada em (4.16) garante que não haverá segmentos vazios em relação à atribuição de indivíduos, isto é, todo o grupo g conterá, mesmo que parcialmente, pelo menos um indivíduo i . Por fim, as restrições dadas em (4.17) e (4.18) impõem, respectivamente, condições de integralidade às variáveis e do modelo e que os valores das variáveis p deverão estar contidos no intervalo $[0, 1]$.

O modelo (4.11–4.18), assim como PPMHLP, é não-linear. Isto se deve à sua função objetivo, na qual há o produto entre as variáveis e e p , e à restrição dada por (4.14), em que há a divisão de termos baseados nas variáveis p . Entretanto, pode-se considerar a reformulação deste problema, assim como apresentado no Capítulo 2, aplicando-se a técnica proposta em [Fortet 1959, Fortet 1960] para o caso em que $m = 2$.

Logo, define-se que o produto $p^{ig} * e_{jk}^g$ será igual uma variável w_{jk}^{ig} , isto é, $w_{jk}^{ig} = p^{ig} * e_{jk}^g$. Este produto é obtido adicionando-se ao modelo três novas restrições que garantam

que $w_{jk}^{ig} = \max(0, p^{ig} + e_{jk}^g - 1)$:

$$w_{jk}^{ig} \leq p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (4.19)$$

$$w_{jk}^{ig} \leq e_{jk}^g, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (4.20)$$

$$w_{jk}^{ig} \geq p^{ig} + e_{jk}^g - 1, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J \quad (4.21)$$

Deve-se notar, a partir da inclusão destas três novas restrições, que os valores das variáveis w_{jk}^{ig} sempre serão zero quando um dos termos, ou p^{ig} ou e_{jk}^g , for zero, dado às restrições em (4.19) e (4.20). No entanto, quando e_{jk}^g tiver valor 1, w_{jk}^{ig} assumirá o valor de p^{ig} , isto é $w_{jk}^{ig} = p^{ig}$, em razão das restrições em (4.19) e (4.21). Esta proposição é verdadeira, pois as variáveis e são binárias.

A representação do produto entre as variáveis p e e por meio das variáveis w permite reescrever a função objetivo dada em (4.11), considerando $m = 2$, como:

$$\text{Minimize } M = \sum_{i=1}^I \sum_{g=1}^G \sum_{j=1}^J \sum_{k=1}^J d_{jk}^i (w_{jk}^{ig})^2, \quad (4.22)$$

pois como as variáveis e_{jk}^g são binárias, temos que $(e_{jk}^g)^2 = e_{jk}^g$. Logo, a seguinte igualdade é válida:

$$(w_{jk}^{ig})^2 = (p^{ig})^2 * e_{jk}^g. \quad (4.23)$$

Prova-se, portanto, que a função objetivo dada por (4.11) é equivalente à função objetivo dada em (4.22) para $m = 2$.

Adicionalmente, pode-se linearizar as restrições dadas em (4.14) pela transferência do denominador situado do lado direito desta desigualdade para o lado esquerdo. Ao realizar esta operação, obtém-se novamente o produto das variáveis p e e . Desta forma, pode-se reescrever esta restrição como

$$\sum_{i=1}^I \sum_{j=1}^J w_{jj}^{ig} \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G. \quad (4.24)$$

Por fim, pode-se adicionar cortes à esta nova formulação, a fim de tornar viável a obtenção de limitantes inferiores quando de sua resolução via algoritmos exatos. Isto é, analogamente à operação realizada para a formulação PPMHLP2, pode-se multiplicar as

restrições dadas em (4.12) por p^{ig} , obtendo-se o seguinte conjunto de novas restrições:

$$\sum_{k=1}^J w_{jk}^{ig} = p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J. \quad (4.25)$$

A partir das definições e reformulações demonstradas, assim como a adição de cortes realizada, sumariza-se o PPMHLP-fuzzy em sua forma quadrática como:

$$\text{Minimize } S = \sum_{g=1}^G \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^J d_{jk}^i (w_{jk}^{ig})^2, \quad (4.26)$$

sujeito a:

$$\sum_{k=1}^J e_{jk}^g = 1, \quad \forall g = 1, \dots, G, \quad j = 1, \dots, J, \quad (4.27)$$

$$\sum_{g=1}^G p_{ig} = 1, \quad \forall i = 1, \dots, I, \quad (4.28)$$

$$w_{jk}^{ig} \leq e_{jk}^g, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (4.29)$$

$$w_{jk}^{ig} \leq p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (4.30)$$

$$w_{jk}^{ig} \geq p^{ig} + e_{jk}^g - 1, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (4.31)$$

$$e_{jk}^g \leq e_{kk}^g, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (4.32)$$

$$\sum_{i=1}^I \sum_{j=1}^J w_{jj}^{ig} \leq \sum_{i=1}^I c^i p^{ig}, \quad \forall g = 1, \dots, G, \quad (4.33)$$

$$\sum_{k=1}^J w_{jk}^{ig} = p^{ig}, \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad (4.34)$$

$$\sum_{i=1}^I p^{ig} \geq 1, \quad \forall g = 1, \dots, G, \quad \forall i = 1, \dots, I, \quad (4.35)$$

$$w_{jk}^{ig} \in [0, 1], \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I, \quad j, k = 1, \dots, J, \quad (4.36)$$

$$e_{jk}^g \in \{0, 1\}, \quad \forall g = 1, \dots, G, \quad j, k = 1, \dots, J, \quad (4.37)$$

$$p^{ig} \in [0, 1], \quad \forall g = 1, \dots, G, \quad i = 1, \dots, I. \quad (4.38)$$

Todas as restrições do modelo (4.26–4.38) já foram explicadas ao longo deste texto, bem como sua função objetivo. Logo, estas não serão descritas novamente. Adicional-

mente, assim como no processo de obtenção da formulação PPMHLP2, pode-se remover do modelo (4.26–4.38) as restrições dadas em (4.30), pois o conjunto de restrições em (4.34) as tornam redundantes.

Isto se deve em razão de que se a soma $\sum_{k=1}^J w_{jk}^{ig}$ deve ter valor igual à p^{ig} , para todo i , g e j . Logo, cada um dos termos do somatório poderá no máximo ter valor igual à p^{ig} . Além disto, dada a restrição em (4.30), apenas um dos termos desta soma poderá ter valor maior que zero, pois as variáveis e_{jk}^g são binárias, implicando que apenas uma das variáveis w desta soma terá exatamente o valor de p^{ig} , e as demais terão valor zero. Esta relação aqui descrita torna obsoletas também as restrições dadas em (4.31).

O modelo (4.26–4.38) representa um problema de otimização quadrático, podendo ser resolvido por meio de *solvers* como o CPLEX. Especialmente para instâncias pequenas, como a utilizada neste capítulo para fins de exemplificação, pode-se obter a solução ótima do problema. Os resultados para esta instância são analisados a seguir, buscando-se verificar quais suposições, ou mesmo conclusões, uma versão *fuzzy* do PPMHLP permitirá analisar.

4.3 Exemplo de aplicação

Nesta Seção são apresentadas algumas considerações em relação aos resultados computacionais do PPMHLP-*fuzzy*, obtidos a partir da solução de seu modelo para o exemplo considerado neste Capítulo. Neste exemplo, supôs-se que três indivíduos participaram de uma tarefa de triagem, categorizando em pilhas quatro restaurantes. Além disto, supôs-se que estes indivíduos qualificaram por meio de uma descrição cada uma de suas pilhas. A estrutura de pilhas de cada indivíduo i é apresentada na Tabela 4.1. Para estes dados, comparam-se os resultados obtidos pela resolução do modelo do PPMHLP, dados na Tabela 4.2, aos resultados obtidos pela resolução do modelo do PPMHLP-*fuzzy*, dados na Tabela 4.3.

Para analisar os resultados, primeiramente, considera-se verificar a segmentação dos indivíduos recuperada. Para o modelo do PPMHLP, obteve-se a alocação dos indivíduos 1 e 3 no segmento 1, indicando que estes indivíduos possuem uma opinião em relação às estruturas de categorias que difere da opinião do indivíduo 2, alocado ao segmento 2. No entanto, sugeriu-se previamente que os três indivíduos estavam em acordo parcial em relação à classificação do restaurante China5 na categoria de comidas típicas. Além disto, sugeriu-se também que os três indivíduos concordavam que o restaurante 100% Brasil está relacionado à venda de refeições tradicionais. Diferentemente do modelo *crisp*, o modelo PPMHLP-*fuzzy* permite observar esta relação de concordância parcial entre os

Segmento 1		Pertinências:	$p^{(1,1)} = 0,8; p^{(2,1)} = 0; p^{(3,1)} = 0,6$
	Descrição	Elementos	
Categoria 1	Comidas Típicas	<i>China5</i>	
Categoria 2	Fast Food	<i>TopMex</i>	Food&Fast
Categoria 3	Refeições	<i>100% Brasil</i>	
Segmento 2		Pertinências:	$p^{(1,2)} = 0,2; p^{(2,2)} = 1; p^{(3,2)} = 0,4$
	Descrição	Elementos	
Categoria 1	Comidas Típicas	<i>China5</i>	TopMex
Categoria 2	Refeições	<i>100% Brasil</i>	Food&Fast

Tabela 4.3: Resultado do PPMHLP-*fuzzy* para o exemplo dado e valores de pertinência $p^{(i,g)}$

três indivíduos em relação à estes dois restaurantes de uma forma mais direta.

Para os resultados dados pelo PPMHLP-*fuzzy*, o indivíduo 2 foi novamente alocado ao segmento 2, com um grau de pertinência $p^{2,2} = 1$. Os indivíduos 1 e 3, por sua vez, foram alocados a este segmento com um grau de pertinência $p^{1,2} = 0,2$ e $p^{3,2} = 0,4$, respectivamente. Para este segmento, o modelo recuperou duas categorias. A Categoria 1 (Comidas Típicas) possui os restaurantes China5 e TopMex. Pode-se, subjetivamente, sugerir que o restaurante China5 é a mediana, o que faria muito sentido, pois os três indivíduos consideraram que este restaurante estava em uma categoria relacionada à comidas típicas. Adicionalmente, os indivíduos 1 e 2 consideraram que apenas este era um restaurante pertencente a esta categoria.

A Categoria 2 (Refeições) deste segmento, engloba os restaurantes 100% Brasil e Food&Fast. Novamente, a definição da mediana pode ser feita de forma subjetiva, pois os três indivíduos colocaram o restaurante 100% Brasil na qualidade de restaurante de refeições. A análise desta categoria permite, ainda, sugerir que a atribuição do indivíduo 3 com maior pertinência que o indivíduo 1 neste segmento está correta, pois o indivíduo 3 além de concordar com o indivíduo 2 em relação à definição do restaurante China5 como de comidas típicas, concordou plenamente que os restaurantes Food&Fast e 100% Brasil estão na mesma categoria (refeições).

Por fim, analisa-se o Segmento 1, para o qual apenas os indivíduos 1 e 3 estão designados ($p^{1,1} = 0,8$ e $p^{3,1} = 0,6$). A estrutura de categorias criada pelo indivíduo 2, de fato, difere fortemente da estrutura de categorias recuperada pelo modelo para o segmento 1. A primeira diferença que pode ser destacada é em relação ao número de categorias. Ambos os indivíduos, 1 e 3, estabeleceram 3 pilhas durante a tarefa de triagem, e o indivíduo 2 estabeleceu apenas duas.

A estrutura de categorias recuperada para este segmento, ainda, é idêntica à estrutura de categorias estabelecida pelo indivíduo 1 durante a tarefa de triagem, o que explica o fato de seu grau de pertinência ser maior para este segmento em relação ao grau de pertinência do indivíduo 3. Em relação às categorias, o Segmento 1 possui em sua Categoria 1 (Comidas Típicas) apenas o restaurante China5, resultado idêntico às estruturas observadas em relação às pilhas dos indivíduos 1 e 3. A Categoria 2 (*Fast Food*), contém os restaurantes TopMex e Food&Fast.

Sugere-se que a escolha do restaurante TopMex como mediana, de forma subjetiva, é apropriada, pois este foi classificado como *fast food* por ambos os indivíduos do segmento. Já o restaurante Food&Fast, foi considerado como de Fast Food apenas pelo indivíduo 1. Por fim, a Categoria 3 (Refeições) contém apenas o restaurante 100%Brasil, julgado como pertinente à esta categoria por ambos os indivíduos. A próxima seção deste texto analisa a proposição desta modelagem parcial para o PPMHLP voltado a ambientes *fuzzy*, ponderando suas implicações, limitações e possibilidades.

4.4 Considerações

Neste capítulo sugeriu-se que o processo de modelagem do PPMHLP pode ser estendido, buscando-se incorporar certo grau de imprecisão ao segmentar os indivíduos. Isto, devido ao fato de que, ao se atribuir um dado indivíduo a apenas um segmento, pode-se assumir que este indivíduo não está de acordo, mesmo que parcialmente, com a opinião dos demais indivíduos alocados a outros segmentos. Isto é, a estrutura de categorias definida por este indivíduo durante a tarefa de triagem difere totalmente da estrutura de categorias recuperada em outros segmentos.

No entanto, o exemplo apresentado na Seção 4.1 mostrou que indivíduos, mesmo que designados a segmentos diferentes, podem categorizar alguns dos objetos de maneira similar. Neste contexto, a utilização de modelos de *cluster* baseados na teoria dos conjuntos *fuzzy* possibilita a representação da ocorrência de similaridades, mesmo que parciais, entre as estruturas de categorias geradas pelos indivíduos durante a tarefa de triagem.

A partir desta motivação, apresentou-se um novo problema a partir do PPMHLP, considerando-se a atribuição dos indivíduos a mais de um segmento. O modelo *fuzzy* gerado, o PPMHLP-*fuzzy*, descrito em (4.11–4.18), tem por objetivo obter um grau de pertinência p^{ig} para cada um dos indivíduos i em relação a todos os segmentos g , além de estabelecer uma estrutura de categorias entre os objetos para cada um dos segmentos.

Este modelo proposto, assim como os modelos *fuzzy* tradicionais, necessita da definição de um parâmetro m , o qual define o nível de imprecisão a ser considerado na

segmentação dos indivíduos. No entanto, seguindo definições encontradas na literatura, mostrou-se que ao definir o valor 2 para este parâmetro, pode-se obter uma formulação quadrática para o problema. Esta formulação, por sua vez, assemelha-se à formulação PPMHLP2, descrita no Capítulo 2. Desta forma, pôde-se eliminar restrições redundantes e adicionar cortes ao modelo.

A formulação obtida, entretanto, não foi testada para instâncias de grande porte, dado que este não é o objetivo principal deste capítulo, nem deste trabalho. Porém, a resolução do PPMHLP-*fuzzy* por meio do *solver* CPLEX, para o exemplo considerado, permitiu analisar a segmentação dos consumidores e as estruturas de categorias recuperadas. Eliminou-se, desta forma, a necessidade de recorrer aos dados referentes às pilhas construídas pelos indivíduos para que se pudesse analisar similaridades de opiniões entre indivíduos, anteriormente designados a segmentos distintos.

Trabalhos futuros envolverão a continuidade desta pesquisa em relação o modelo do PPMHLP-*fuzzy*. Serão buscadas formas de melhorar o novo modelo sugerido, bem como verificar o quão eficiente este poderá ser em termos de resolução exata. A partir disto, um novo algoritmo poderá ser formulado a fim de possibilitar sua aplicação a problemas de grande porte, como o exemplo de aplicação mostrado para o PPMHLP no Capítulo 3.

Capítulo 5

Considerações finais

Apresentou-se neste trabalho um novo modelo para o problema das p -medianas heterogêneo, proposto por [Blanchard et al. 2012]. Este novo modelo consiste em um problema de otimização não paramétrico, exceto pelo número de segmentos G , no qual eliminou-se o fator de penalidade δ do modelo original. Este parâmetro foi inicialmente introduzido por Blanchard et al. (2012) em seu modelo para automatizar a recuperação da estrutura de categorias em cada um dos segmentos considerados. Adicionalmente, duas formulações lineares para o PPMHLP foram apresentadas. A resolução destas formulações via algoritmos exatos tradicionais, por sua vez, possibilitou a obtenção de limitantes inferiores para este novo problema, permitindo validar a metaheurística proposta neste trabalho para sua resolução de forma eficiente.

Os resultados apresentados neste trabalho não se limitam somente à área acadêmica. Tomadores de decisão atuantes nos mais diversos segmentos, principalmente àqueles relacionados a áreas como Gestão e Marketing, poderão se beneficiar a partir de sua aplicação. O modelo PPMHLP e seu algoritmo, o VNS-PPMHLP, permitem que se compreenda como um produto, ou mesmo uma empresa, é percebido por seus clientes em relação aos seus concorrentes. Porém, o grande diferencial deste trabalho está relacionado à melhoria do modelo do PPMH, proposto por Blanchard et al. (2012), bem como à obtenção de um algoritmo mais preciso e robusto que o VNS-PPMH, algoritmo também apresentado pelos autores.

Esta abordagem genuína dada a problemas de *clustering* por Blanchard et al. (2012) é extremamente relevante, pois permitiu tratar de forma diferenciada a heterogeneidade, fator que se faz presente sempre que há um grupo de indivíduos manifestando sua opinião. No entanto, o modelo apresentado pelos autores se mostra um tanto complexo aos olhos de seus possíveis usuários, bem como aos olhos de especialistas em modelos de otimização combinatória. Estes últimos, por sua vez, estão intimamente ligados à academia e à Pesquisa Operacional.

O primeiro fator de complicação apresentado aos usuários pelo modelo original de Blanchard et al. (2012) é a definição de seus parâmetros. Ao necessitar da fixação de um fator de penalidade δ para que o modelo possa recuperar a informação desejada, fica subentendido que o usuário necessitará deter amplo conhecimento do problema, bem como de seus dados. Apesar de Blanchard et al. (2012) sugerirem como apropriado o uso de regras simples para a definição deste parâmetro, os resultados obtidos neste trabalho mostram que a precisão de seu modelo é inferior à precisão do novo modelo proposto.

Sendo assim, a proposição do modelo PPMHLP, bem como do algoritmo VNS-PPMHLP, surge como uma alternativa menos complexa ao processo decisório. Destaca-se como diferencial a forma não paramétrica do modelo apresentado, exceto pela definição do número de segmentos G a serem identificados. No entanto, a definição deste parâmetro pode sim ser realizada de forma simples, como mostrado, bastando observar o comportamento da função objetivo. Ficou evidente, portanto, que um conhecimento avançado em relação a modelos de otimização combinatória não se faz necessário para a utilização deste modelo, o que beneficia qualquer possível usuário final.

A eliminação do fator de penalidade possibilitou também a obtenção de uma formulação linear para o problema, a partir da qual se pôde observar limitantes inferiores. Isto permitiu não só validar o algoritmo proposto para o PPMHLP em termos de desempenho, como demonstrar que este novo modelo gera resultados de melhor qualidade no que diz respeito à recuperação da informação se comparado ao modelo de Blanchard et al. (2012). Ao observar a qualidade das soluções obtidas em termos de custo, comparando-as a limitantes inferiores, pôde-se ter maior segurança ao interpretar os resultados do algoritmo proposto, diferentemente do VNS-PPMH. Para este algoritmo não há qualquer indício de que as soluções obtidas estejam próximas às soluções ótimas.

Numericamente, observou-se considerável superioridade em termos de precisão do algoritmo VNS-PPMHLP e do modelo PPMHLP em relação ao VNS-PPMH e PPMH. Por exemplo, ao recuperar a estrutura de categorias disposta entre os objetos durante o processo de simulação de Monte Carlo, para as 27 instâncias analisadas o VNS-PPMHLP supera o VNS-PPMH em 14 casos. Em outros 9 casos, os algoritmos são equivalentes. Ainda, o ARI médio de 0,95 obtido pelo VNS-PPMHLP neste quesito indica que este algoritmo realizou uma recuperação quase perfeita das informações.

Adicionalmente, ao recuperar a segmentação dos indivíduos para estas instâncias, o novo algoritmo apresentou um grau de precisão ligeiramente maior que o algoritmo de Blanchard et al. (2012), que também recuperou de forma satisfatória esta informação. Estatisticamente, no entanto, esta diferença não foi significativa. Porém, o estudo de caso mostrado neste trabalho revelou que a segmentação dos indivíduos gerada pelos algorit-

mos VNS-PPMHLP e VNS-PPMH difere radicalmente. Apesar de ambos os algoritmos indicarem que dois segmentos deveriam ser considerados entre os indivíduos, observou-se um ARI de 0,34 ao se comparar a segmentação produzida pelos dois algoritmos. Esta composição diferenciada na segmentação dos indivíduos, no entanto, fez com que o VNS-PPMHLP tivesse uma precisão 7% maior que o VNS-PPMH ao recuperar a estrutura de categorias para estes dados.

Pode-se atribuir esta superioridade do VNS-PPMHLP, em grande parte, ao novo modelo proposto neste trabalho, o PPMHLP. Este modelo de fato reduziu a chance de se cometerem erros ao configurar o algoritmo de resolução. Não há como saber se o desempenho inferior do VNS-PPMH foi devido à escolha do fator de penalidade, a não ser que grande esforço computacional seja destinado à esta investigação. Esta investigação, por sua vez, não faria sentido, pois o valor ideal ao qual se chegaria para o fator de penalidade estaria restrito apenas às instâncias utilizadas nesta investigação. Isto é, o modelo não poderia ser configurado de forma universal. Diferentemente, o PPMHLP mostra-se como um modelo bastante genérico, podendo ser aplicado à qualquer conjunto de dados sem a necessidade de um ajuste de parâmetros específico para estes dados.

Outro ponto a ser discutido acerca do VNS-PPMHLP é a sua confiabilidade. Este algoritmo demonstrou menor sensibilidade do que o VNS-PPMH às diversas características presentes nas instâncias. Ao mostrar esta consistência, associada às soluções de boa qualidade em termos de recuperação da informação, têm-se uma maior segurança quanto à interpretação dos resultados do VNS-PPMHLP. Contrariamente, por exemplo, o algoritmo VNS-PPMH se mostra significativamente sensível ao resolver instâncias em que há valores heterogêneos para as matrizes de dissimilaridades. Isto é, para casos reais, as matrizes de dissimilaridades observadas para os diferentes indivíduos tendem a possuir valores diferenciados se comparadas entre si.

Para estes casos, no entanto, o algoritmo VNS-PPMHLP se mostrou robusto, garantindo constância no nível de qualidade de suas soluções. Esta conclusão é baseada na análise dos dados obtidos a partir da Simulação de Monte Carlo, especialmente para aquelas instâncias em que se adicionou alto grau de ruído às matrizes de dissimilaridades. De fato, a conclusão sugerida nesta análise é verificada em termos práticos, pois como destacado anteriormente o VNS-PPMHLP foi mais preciso ao recuperar a estrutura de categorias para os dados reais considerados. Desta forma, acredita-se que os resultados apresentados neste trabalho ajudarão a diminuir a distância entre teoria e prática, uma vez que se produziu uma ferramenta com menor complexidade de uso e que é, de fato, mais confiável do que a tida até então. Quanto às implicações em relação à área acadêmica, os resultados mostrados neste trabalho sugerem ainda inúmeras possibilidades.

Propôs-se a reformulação do PPMHLP para uma forma linear, consistindo esta em uma problema de programação linear inteira mista. Obtiveram-se, para tal, duas formulações distintas, a PPMHLP1 e a PPMHLP2. Ambas puderam ser testadas via resolvedores comerciais, utilizando-se de algoritmos exatos tradicionais. Permitiu-se, assim, verificar limitantes inferiores para o PPMHLP, podendo-se provar que o VNS-PPMHLP encontrou a solução ótima em duas das instâncias testadas e que este algoritmo chegou muito próximo do ótimo global em relação a outras 6 instâncias, com GAP inferior a 5%.

Blanchard et al. (2012) em seu trabalho não forneceram quaisquer informações acerca do ótimo global em relação ao PPMH. Neste trabalho, diferentemente, pôde-se avaliar o quão apropriados algoritmos de *branch-and-cut* são para a resolução do problema proposto. Mesmo não obtendo total êxito na aplicação destes algoritmos, utilizando-se uma das ferramentas comerciais mais relevantes neste segmento, esta limitação observada garante que continuar esta pesquisa, desenvolvendo-se algoritmos especializados para a resolução exata do problema, será uma contribuição de grande relevância. Pode-se ainda investigar outros algoritmos aproximados para a resolução do PPMHLP, pois neste trabalho apenas a metaheurística VNS foi testada.

Adicionalmente, a extensão do PPMHLP a ambientes *fuzzy*, sugerida neste trabalho, poderá motivar ainda mais a continuidade desta pesquisa no âmbito acadêmico. O PPMHLP é um modelo que trata de forma diferenciada a heterogeneidade, o que o torna contrastante com os muitos modelos clássicos de *clustering*, gerando novas possibilidades em termos de análise. No entanto, uma versão *fuzzy* pode estender ainda mais estas possibilidades, fornecendo diversas informações complementares, como demonstrado no Capítulo 4 deste texto. Por exemplo, pode-se observar que indivíduos que, embora devam ser associados a segmentos distintos, por se considerar que estes têm opiniões divergentes, podem estar parcialmente de acordo sob alguns aspectos.

Além desta possibilidade em termos de análise de resultados, o desenvolvimento de um modelo *fuzzy* para PPMHLP será uma tarefa complexa. A formulação apresentada neste trabalho possui uma função objetivo quadrática. Assim como as formulações apresentadas para o PPMHLP, esta é resolvida de forma menos eficiente por algoritmos exatos tradicionais. Logo, duas linhas podem ser seguidas. A primeira delas é trabalhar na modificação e melhora desta formulação. A outra, por sua vez, é trabalhar no sentido de desenvolver um algoritmo exato apropriado para este novo problema.

Referências Bibliográficas

- Aldenderfer, M. & R. Blashfield (1984), *Cluster Analysis*, Sage Publications, Beverly Hills, USA.
- Aloise, D. & P. Hansen (2009), Clustering, *em* D.Shier, ed., ‘Handbook of Discrete and Combinatorial Mathematics’, CRC Press.
- Anderberg, M. (1973), *Cluster Analysis for Applications*, Academic, New York.
- Andreas, Pierre, Stefan & Timo (2014), Couenne, an exact solver for nonconvex minlps, Relatório técnico, IBM and Carnegie Mellon University.
URL: <https://projects.coin-or.org/Couenne>
- Applegate, D. L., R. E. Bixby, C. Chvatal & W. J. Cook (2007), *The traveling salesman problem: a computational study*, Princeton University Press.
- Arenales, M., V. A. Armentano, R. Morabito & H. H. Yanasse (2006), *Pesquisa Operacional*, Editora Campus.
- Bettman, J. R., M. F. Luce & J. W. Payne (1998), ‘Constructive consumer choice processes’, *Journal of Consumer Research* **25**(3), 187–217.
- Bettman, James. R. Bettman & Whan Park (1980), ‘Effects of prior knowledge and experience and phase of the choice process on consumer decision processes: A protocol analysis’, *Journal of Consumer Research* (7), 234–248.
- Blanchard, Simon J., & Wayne S. DeSarbo (2013), ‘A new zero-inflated negative binomial methodology for latent category identification’, *Psychometrika* **78**, 322–340.
- Blanchard, Simon J., Daniel Aloise & Wayne S. DeSarbo (2012), ‘The heterogeneous p-median problem for categorization based clustering’, *Psychometrika* **77**(4), 741–762.
- Brimberg, Jack, Pierre Hansen, Nenad Mladenović & Eric D. Taillard (2000), ‘A new zero-inflated negative binomial methodology for latent category identification’,

Operations Research **48**(3), 444–460.

URL: <http://pubsonline.informs.org/doi/abs/10.1287/opre.48.3.444.12431>

Brusco, M. J., D. Steinley, J. D. Cradit & R. Singh (2012), ‘Emergent clustering methods for empirical om research’, *Journal of Operations Management* **30**(6), 454–466.

Brusco, M. J. & H. F. Kohn (2009), ‘The analysis of free-sorting data: beyond pairwise co-occurrence’, *Psichometrika* **74**(3), 457–475.

Chen, M. S., J. Han & P. S. Yu (1999), ‘Data mining: an overview from a database perspective’, *IEEE Transactions on Knowledge and data Engineering* **8**(6), 866–883.

Courcoux, Ph., P. Faye & E. M. Qannari (2014), ‘Determination of the consensus partition and cluster analysis of subjects in a free sorting task experiment’, *Food Quality and Preference* **32**, 107–112.

Coxon, A. P. M. (1999), *Sorting data: collection and analysis*, Thousand Oaks: Sage.

Dantzig, G. B. (1963), *Linear Programming and Extensions*, Princeton University Press, Princeton, NJ.

Daws, J. T. (1996), ‘The analysis of free-sorting data: beyond pairwise co-occurrence’, *Journal of Classification* **13**(1), 57–80.

Duda, R. & P. Hart (1973), *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, USA.

Dunn, J. C. (1973), ‘A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters’, *Journal of Cybernetics* **3**(3), 32–57.

Fisher, R. A. (1936), ‘The use of multiple measurements in taxonomic problems’, *Annals of Eugenics* **7**, 179–188.

Forgy, E. W. (1965), ‘Cluster analysis of multivariate data: efficiency vs. interpretability of classifications’, *Biometrics* **21**(3), 768–769.

Fortet, R. (1959), ‘L’algèbre de boole et ses applications en recherche opérationnelle’, *Cahiers du Centre d’Études de Recherche Opérationnelle* **1**(4), 5–36.

Fortet, R. (1960), ‘Applications de l’algèbre de boole en recherche opérationnelle’, *Revue Française d’Informatique et de Recherche Opérationnelle* **4**(14), 17–26.

- Ghiani, G., G. Laporte & R. Musmanno (2004), *Introduction to Logistics Systems Planning Control*, Wiley.
- Griffin, A. & J. R. Hauser (1993), 'The voice of customer', *Marketing Science* **12**(1), 1–27.
- Han, J. & M. Kamber (2000), *Data Mining: concepts and techniques*, Morgan Kaufmann Publishers, San Mateo, CA.
- Hansen, Pierre, Jack Brimberg, , Dragan Urošević & Nenad Mladenović (2009), 'Solving large p-median clustering problems by primal-dual variable neighborhood search', *Data Min Knowl Disc* **19**, 351–375.
- Hansen, Pierre & Nenad Mladenović (1997), 'Variable neighborhood search for the p-median', *Location Science* **5**(4), 207–226.
- Hansen, Pierre & Nenad Mladenović (2001), 'Variable neighborhood search: Principles and applications', *European journal of operational research* **130**(3), 449–467.
- Hansen, Pierre & Nenad Mladenović (2008a), 'Complement to a comparative analysis of heuristics for the p-median problem', *Statistics and Computing* **1**(18), 41–46.
- Hansen, Pierre & Nenad Mladenović (2008b), 'Complement to a comparative analysis of heuristics for the p-median problem', *Stat Comput* **18**, 41–46.
- Hansen, Pierre, Nenad Mladenović & José A. Moreno Pérez (2010), 'Variable neighbourhood search: methods and applications', *Annals of Operations Research* **175**(1), 367–407.
- Höppner, F., F. Klawonn, R. Kruse & T. Runkler (1999), *Fuzzy Cluster Analysis*, Wiley, Chichester, UK.
- Hubert, Lawrence & Phipps Arabie (1985), 'Comparing partitions', *Journal of Classification* **2**, 192–215.
- IBM (2014), Ibm cplex optimizer, Relatório técnico, IBM.
URL: <http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/>
- Isen, A. M. (2012), *Toward understanding the role of affect in cognition*, Hillsdale: Lawrence Erlbaum.

- John, D. R. & M. Sujan (1990), 'Age differences in product categorization', *Journal of Consumer Research* **16**, 452–460.
- Johnson, S. C. (1965), 'Hierarchical clustering schemes', *Psychometrika* **32**(3), 241–254.
- Jones, D. F., S. K. Mirrazavi & M. Tamiz (2002), 'Multi-objective meta-heuristics: an overview of the current state-of-art', *European Journal of Operations Research* **137**, 1–9.
- Kariv, O. & S. L. Hakimi (1979), 'An algorithmic approach to location problems, part ii: p-medians', *Journal of Applied Mathematics* **37**, 539–560.
- Kelter, S., R. Cohen, D. Engel, G. List & H. Stronher (1977), 'The conceptual structure of aphasic and schizophrenic patients in a nonverbal sorting task', *Journal of Psycholinguistic Research* **6**(4), 279–303.
- Kleinberg, Jon (2002), 'An impossibility theorem for clustering', *Advances in Neural Information Processing Systems* **15**.
- Kohn, H. F., D. Steinley & M. J. Brusco (2010), 'The p-median model as a tool for clustering psychological data', *Psychological Methods* **15**(1), 87–95.
- Land, A. H. & A. G. Doig (1960), 'An automatic method for solving discrete programming problems', *Econometrica* **28**, 497–520.
- Lee, K. & Z Geem (2005), 'A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice', *Computer methods in applied mechanics and engineering* **194**, 618–641.
- Linden, G., B. Smith & J. York (2003), 'Amazon.com recommendations: item-to-item collaborative filtering', *IEEE Internet Computing* **7**, 76–80.
- Liu, Huan, & Lei Yu (2005), 'Toward integrating feature selection algorithms for classification and clustering', *IEEE Transactions on Knowledge and Data Engineering* **17**(4), 491–502.
- Medin, D. L. & M. M. Schaffer (1978), 'Context theory of classification learning', *Psychological Review* **85**(3), 207–238.
- Miller, G. A. (1969), 'A psychological method to investigate verbal concepts', *Journal of Mathematical Psychology* **6**(2), 169–191.

- Mladenović, Nenad, Jack Brimberg, Pierre Hansen & Jose A. Moreno-Pérez (2007), 'The p-median problem: A survey of metaheuristic approaches', *European Journal of Operational Research* **179**, 927–939.
- Mladenović, Nenad & Pierre Hansen (1997), 'Variable neighborhood search', *Computers & Operations Research* **24**(11), 1097–1100.
- Perkins, W. S. (1993), 'The effects of experience and education on the organization of marketing knowledge', *Psychology & Marketing* **10**(3), 169–183.
- Resende, M.G. C. & Renato F. Werneck (2004), 'A hybrid heuristic for the p-median problem', *Journal of Heuristics* **10**, 59–88.
- Ross, B. H. & G. L. Murphy (1999), 'Food for thought: cross-classification and category organization in a complex real-world domain', *Cognitive Psychology* **38**(4), 495–554.
- Schafer, J. B., J. A. Konstan & J. E. Riedl (2001), 'E-commerce recommendation applications', *Data Mining and Knowledge Discovery* **5**, 115–153.
- Sherali, Hanif D. & Amine Alameddine (1992), 'A new reformulation-linearization technique for bilinear programming problems', *Journal of Global Optimization* **2**(4), 379–410.
- Sherali, Hanif D. & Jitendra Desai (2005), 'A global optimization rlt-based approach for solving the fuzzy clustering problem', *Journal of Global Optimization* **33**, 597–615.
- Shugan, S. M. (1980), 'The cost of thinking', *Journal of Consumer Research* **7**(2), 99–111.
- Simon, H. A. (1955), 'A behavioral model of rational choice', *Quarterly Journal of Economics* **69**(1), 99–118.
- Siridov, Denis, W. S. Wei, Igor Vasilyev & Saverio Salerno, eds. (2008), *Automatic Defects Classification with p-median Clustering Technique*, Hanoi, Vietnam.
- Sujan, M. & C. Dekleva (1987), 'Product categorization and inference making: some implications for comparative advertising', *Journal of Consumer Research* **14**(3), 372–378.
- Takane, Y. (1980), 'Analysis of categorizing behavior using a quantification method', *Behaviormetrika* **8**(7), 75–86.

- Urban, G.L., J. S. Hulland & B. D. Weinberg (1993), 'Premarket forecasting for new consumer durable goods: modeling categorization, elimination, and consideration phenomena', *Journal of Marketing* **57**(2), 47–63.
- Viswanathan, Madhubalan, Michael D. Johnson & Seymour Sudman (1999), 'Understanding consumer usage of product magnitudes through sorting tasks', *Psychology & Marketing* **16**(8), 643–657.
- Ward, J. H. (1963), 'Hierarchical grouping to optimize an objective function', *Journal of the American Statistical Association* **30**(301), 236–244.
- Xu, Rui & Donald Wunsch (2005), 'Survey of clustering algorithms', *IEEE Transactions on Neural Networks* **16**(3).