

**COMBINANDO BUSCA TABU E ALGORITMOS GENÉTICOS PARA RESOLVER
O PROBLEMA DE MARKETING DIRETO CONSIDERANDO O CANIBALISMO
ENTRE PRODUTOS**



<https://doi.org/10.56238/arev6n4-413>

Data de submissão: 26/11/2024

Data de publicação: 26/12/2024

Felipe Martins Müller
Universidade Federal de Santa Maria/PPGEP
E-mail: felipe@inf.ufsm.br

Vanessa Andréia Schneider
Universidade Federal de Santa Maria/PPGEP
E-mail: vanessaandreias@gmail.com

Iaê Santos Bonilha
Universidade Federal de Santa Maria/PPGEP
E-mail: iaesb@hotmail.com

Veridiane Bárbara de Souza
Universidade Federal de Santa Maria/PPGEP
E-mail: veridianes@hotmail.com

Greici da Rosa da Cruz
Universidade Federal de Santa Maria/PPGCC
E-mail: inf.greici@gmail.com

Michalis Mavrovouniotis
Centro de Excelência em Pesquisa e Inovação KIOS
Departamento de Engenharia Elétrica e de Computação, Universidade de Chipre, 2109 Nicósia,
Chipre.
E-mail: mavrovouniotis.michalis@ucy.ac.cy

Gédson Mário Borges Dal Forno
Universidade Federal de Santa Maria / Departamento de Computação Aplicada
Centro de Tecnologia, Prédio 07, 97105-900, Santa Maria-RS, Brasil
E-mail: gedsondf@gmail.com

RESUMO

Este trabalho trata do problema da seleção de um conjunto de clientes que receberão uma oferta para um ou mais produtos durante uma campanha de promoção. Essas campanhas são ferramentas de marketing fundamentais para melhorar o lucro econômico de uma empresa, seja pela aquisição de novos clientes ou pela geração de receita adicional de clientes existentes. Trabalhamos com um modelo matemático bem conhecido para o problema. Acrescentamos ao problema a restrição do canibalismo, que evita que alguns produtos sejam oferecidos simultaneamente, para simular produtos concorrentes canibalizando o mercado uns dos outros. Propomos uma heurística híbrida, a primeira combinando um Algoritmo Genético (GA) com Tabu Search (TS). Extensos experimentos computacionais foram realizados em um conjunto de problemas de teste da literatura com e sem a restrição de canibalismo.

Comparamos nosso método com um TS e um Matheuristic da literatura. O método híbrido supera os métodos concorrentes em todos os casos de teste e em todos os tamanhos de instâncias.

Palavras-chave: Problema de marketing direto. Heurística híbrida. Canibalismo.

1 INTRODUÇÃO

Recentemente, a Mintel, uma das principais agências de inteligência de mercado do mundo, divulgou um relatório propondo sete tendências de clientes que moldarão os mercados globais nos próximos 10 anos. Pode-se resumir essas tendências como 1) Bem-estar: Buscando o bem-estar físico e mental; 2) Entorno: Sentir-se conectado ao ambiente externo; 3) Tecnologia: Encontrar soluções por meio da tecnologia nos mundos físico e digital; 4) Direitos: Sentir-se respeitado, protegido e apoiado; 5) Identidade: Compreender e expressar-se em sociedade; 6) Valor: Encontrar benefícios tangíveis e mensuráveis dos investimentos; e 7) Experiências: Buscando e descobrindo estímulos. (Crabbe et al. 2019). Considerando esses aspectos, é preciso pensar em como alcançar e manter uma carteira de clientes oferecendo bons produtos no momento certo. Para alcançar esse recurso, vários bancos de dados de consumidores precisam ser explorados. Esses bancos de dados contêm informações robustas sobre consumidores e mercados, econômicos, demográficos, tecnológicos, políticos e sociológicos que ajudam os analistas a tomar suas decisões.

As campanhas promocionais são uma das ferramentas fundamentais de marketing mais direto para aquisição de clientes e geração de lucro geral (Kotler e Armstrong, 2016), (Abedi, 2017), (Nobibon et al. 2011), (Ładyzynski et al., 2019).

As campanhas promocionais geralmente visam os clientes considerando fatores como a probabilidade de resposta positiva, lucro projetado, custo projetado da oferta individual e saturação excessiva do cliente. Todos esses fatores geram problemas combinatórios difíceis de resolver (Praag, 2010) e merecem atenção da comunidade acadêmica (Czikosova, et al., 2014). Um conjunto muito bom de referências relacionadas ao problema foi apresentado por (Cetin e Alabas-Uslu, 2015), e apresentaremos novas referências aqui.

Apesar das poucas referências de heurísticas para resolver o problema do marketing direto, elas tendem a ser abordagens promissoras como substitutas dos métodos estatísticos (Cohen, 2004), (Nobibon et al. 2011). Alguns exemplos podem ser encontrados em (Bhaskar et al., 2009) que propõe o uso de lógica difusa para seleção de clientes em uma campanha de marketing de venda cruzada de um banco. (Nobibon et al. 2011) propõe uma formulação matemática para o problema, bem como duas abordagens principais para resolvê-lo: uma abordagem exata e uma abordagem heurística baseada na busca tabu, que teve um desempenho muito bom em instâncias pequenas e grandes. Eles também disponibilizaram um conjunto de instâncias e limites para o problema que usaremos para testar e comparar os métodos propostos neste artigo. (Oliveira et al., 2015) apresentaram um esquema híbrido com GRASP e VNS. (Cetin e Alabas-Uslu, 2015) propuseram uma abordagem diferente que divide o problema em dois problemas de decisão: atribuir produtos à campanha de mercado e atribuir ofertas à

base de clientes. Os problemas de decisão foram resolvidos com programação linear e uma conexão heurística entre ambos, obtendo bons resultados para todos os tipos de instâncias. No entanto, como ele usa um procedimento exato, há limitações quando o tamanho do problema aumenta e mostramos aqui que existem alguns casos em que esses procedimentos não são capazes de resolver.

Recentemente (Souza, 2018) e (Müller et al. 2019) propuseram dois esquemas híbridos combinando GRASP e Algoritmo Genético (GA) com Tabu Search, respectivamente. Então, de acordo com nosso conhecimento, pela primeira vez, a restrição do canibalismo foi considerada no problema do marketing direto (Souza, 2018), e um novo conjunto de problemas de teste foi gerado e testado heuristicamente. O canibalismo pode ser entendido como o efeito que um produto exerce sobre outro, tornando-o sem atratividade, ou seja, se um cliente comprar um, não comprará o outro. Mais tarde (Schneider, 2019) adaptou um modelo matemático incluindo algumas restrições inteligentes para resolver exatamente quase todas as instâncias geradas, incluindo o canibalismo. (Coelho et al., 2017) estendeu a VNS proposta por (Oliveira et al., 2015), para uma versão bi-objetiva do problema baseada nos conceitos de maximização de lucros e busca, ao mesmo tempo, de um conjunto de clientes com menor variabilidade sobre o retorno esperado.

Neste trabalho, apresentamos uma Busca Tabu com um Algoritmo Genético embarcado para lidar com o canibalismo, denominado TSeGA. Testes extensivos foram realizados no mesmo conjunto de instâncias originalmente proposto por (Nobibon et al. 2011) e também propusemos um novo e mais difícil conjunto de instâncias muito grandes e comparamos o TSeGA com dois procedimentos implementados.

Em resumo, fizemos as seguintes contribuições: 1) propusemos o TSeGA, uma nova forma de híbrido de TS e GA; 2) incluímos a restrição de canibalismo e adaptamos todos os procedimentos heurísticos e exatos para abordá-la; 3) mostramos que os métodos heurísticos propostos por (Cetin e Alabas-Uslu, 2015) não são robustos porque não são capazes de resolver algumas das instâncias propostas; e 4) apresentamos que o TSeGA resolve todas as instâncias propostas e tem um desempenho melhor do que os outros métodos da literatura.

O restante deste artigo está organizado da seguinte forma. Na seção 2, apresentamos a descrição do problema, seu modelo matemático e uma breve descrição dos outros procedimentos heurísticos em comparação com o TSeGA. A Seção 3 detalha a heurística híbrida TSeGA e todos os procedimentos relacionados a ela. A seção 4 apresenta os resultados dos cálculos e a última seção conclui o artigo e mostra possíveis caminhos para pesquisas futuras.

2 DESCRIÇÃO DO PROBLEMA

O problema de marketing direto pode ser dividido em dois problemas de decisão, o primeiro decidindo quais produtos participarão da campanha promocional. Lembre-se de que as campanhas promocionais geralmente são focadas em um grupo de clientes e devem ser adaptadas para evitar saturar os clientes com ofertas, portanto, a escolha de quais produtos serão incluídos na campanha deve ter um forte impacto nos resultados da campanha. O segundo problema de decisão é quando os clientes recebem ofertas dos produtos que estão participando da campanha promocional. Essas duas decisões podem ser representadas por duas variáveis binárias: $y_j = \{0 \text{ ou } 1\}$, que indica se o produto j está participando da campanha de mercado ou não; e, $x_{ij} = \{0 \text{ ou } 1\}$, que indica se o produto j está sendo oferecido ao cliente i ou não. Para ambas as variáveis, um valor de 1 indicará a afirmativa e um valor de 0 indicará a negativa.

Neste artigo, o modelo de problema proposto por (Nobibon et al., 2011) foi adaptado para lidar com o canibalismo, conforme mostrado em (Schneider, 2019). O problema do marketing direto é composto por dois elementos principais: o conjunto de clientes C e o conjunto de produtos P . Cada produto $j \in P$ tem um orçamento; um custo fixo f_j que é o custo fixo do produto j que participa da campanha; e uma cota de oferta O_j , que é o número mínimo de clientes que devem receber a oferta para que sua participação na campanha de marketing seja justificável. Cada cliente $i \in C$ tem um lucro projetado para cada oferta de um produto j ; um custo c_{ij} associado a cada oferta de um produto j a um cliente i ; um lucro potencial líquido $NPP_{ij} = p_{ij} / c_{ij}$, que representa o retorno por unidade monetária investida em uma oferta do produto j ao cliente i ; e um limite M_i , que é o limite colocado para simular a saturação da oferta que pode resultar na resposta negativa dos clientes em relação à campanha. Em seguida, o modelo matemático pode ser definido a partir das equações (1) a (8).

<i>Maximizar</i>	$\sum_{i=1}^m \sum_{j=1}^n (p_{ij} - c_{ij})x_{ij} - \sum_{j=1}^n f_j y_j$	(1)
<i>Assunto A:</i>	$\sum_{i=1}^m \sum_{j=1}^n p_{ij}x_{ij} \geq (1 + R) [\sum_{i=1}^m \sum_{j=1}^n c_{ij}x_{ij} + \sum_{j=1}^n f_j y_j]$	(2)
	$\sum_{i=1}^m c_{ij}x_{ij} \leq B_j y_j, j = 1, \dots, n,$	(3)
	$\sum_{j=1}^n x_{ij} \leq M_i, i = 1, \dots, m,$	(4)
	$\sum_{i=1}^m x_{ij} \geq O_j y_j, j = 1, \dots, n,$	(5)

	$\sum_{i=1}^m x_{ij} \leq m \cdot y_j, j = 1, \dots, n,$	(6)
	$y_i + y_j \leq 1 \forall (i, j) \in Can$	(7)
	$y_j, x_{ij} \in \{0,1\} i = 1, \dots, m, j = 1, \dots, n.$	(8)

(7)

O principal objetivo do problema de marketing direto é maximizar o lucro líquido gerado por todas as ofertas feitas na campanha, veja a equação (1). Para ser considerada viável, a campanha de mercado deve atingir a taxa mínima R , que é o retorno por ponto monetário investido na campanha, ver equação (2). A equação (3) representa que o orçamento máximo por produto é respeitado e a inclusão da variável 0-1 y_j no lado direito da equação (3) acelera a execução do CPLEX. A equação (4) limita o número máximo de ofertas por cliente, enquanto a equação (5) define que, se um produto j participar da campanha, pelo menos $O_j > 0$ clientes receberão uma oferta, e a equação (6) especifica que, quando um produto não fizer parte da campanha, nenhum cliente receberá uma oferta. A equação (7) define quais produtos são mutuamente exclusivos, ou seja, se o par de índices de produtos pertence ao conjunto Can , significa que eles não podem ser oferecidos aos clientes na mesma campanha. Finalmente, o último conjunto de restrições (8) são as restrições de integralidade. Observe que, se a equação (7) for retirada, o modelo é o mesmo que (Nobibon *et al.*, 2011).

(Nobibon *et al.*, 2011) apresentaram sete heurísticas e um algoritmo de busca tabu, que superam os métodos usados na época. Eles também fornecem um conjunto de instâncias e seu valor objetivo para as soluções ideais encontradas até agora por um algoritmo de ramificação e preço. O procedimento heurístico apresentado foi direto e, portanto, a pesquisa tabu foi reimplementada por nós para fins de comparação, uma vez que o código-fonte não estava disponível. Detalhes desses procedimentos podem ser encontrados no artigo original.

(Cetin e Alabas-Uslu, 2015) apresentaram duas heurísticas para resolver o problema, nas quais determinam os produtos a serem incluídos em uma campanha usando regras heurísticas e, em seguida, distribuem esses produtos aos clientes de maneira otimizada. A estratégia é realizada em duas fases. Na fase I, um modelo de programação linear é utilizado para prever quais produtos são selecionados ou removidos da campanha do produto. Depois que os produtos da campanha são determinados, o problema de segmentação do produto é reduzido a um tipo de problema de atribuição generalizado. Na fase II, os produtos selecionados na fase I são distribuídos aos clientes de forma otimizada por outro modelo de otimização. As duas fases são conectadas por meio de uma regra heurística. Duas regras heurísticas alternativas, derivadas da programação linear proposta na fase I, chamadas H-R1 e HR-2, são sugeridas para prever os produtos eliminados da campanha (ou equivalentemente os produtos envolvidos na campanha).

Essas regras heurísticas geram dois procedimentos diferentes que superam os resultados apresentados por (Nobibon *et al.*, 2011), mas como usa solucionadores comerciais para executar o LP associado, ele tem uma capacidade limitada na dimensão do problema a ser tratado, e alguns casos são insolúveis por esses métodos. Esta última questão não é comentada no artigo original, mas concebemos alguns exemplos neste artigo que mostram esse problema.

(Oliveira *et al.*, 2015) propuseram um algoritmo heurístico híbrido, GRASP/VNS para resolver o problema. Seu procedimento supera os resultados de (Nobibon *et al.*, 2011), mas é superado pelos resultados de (Cetin e Alabas-Uslu, 2015). Acreditamos que, como sua pesquisa foi publicada no mesmo ano, eles trabalharam simultaneamente.

Na seção 3, apresentamos uma abordagem híbrida inovadora, denominada TSeGA, que resolve todas as instâncias propostas por (Nobibon *et al.*, 2011) e um novo conjunto de instâncias aqui apresentadas e alcança resultados muito melhores do que as outras heurísticas propostas neste artigo. O TSeGA proposto é capaz de superar as heurísticas existentes, conforme mostrado na seção 4.

3 HEURÍSTICA HÍBRIDA TSEGA

TSeGA é uma pesquisa Tabu com um algoritmo genético incorporado. Todos os procedimentos desenvolvidos foram apresentados em detalhes nesta seção.

3.1 ALGORITMOS CONSTRUTIVOS

Para construir a população inicial de AG foram utilizados três algoritmos construtivos diferentes: dois determinísticos e um com randomização. O primeiro algoritmo construtivo é baseado no procedimento de solução inicial para o algoritmo Tabu Search proposto em (Nobibon *et al.*, 2011) com modificações no procedimento de seleção de produtos; o segundo é uma versão aleatória gananciosa do primeiro e; o terceiro começa com uma solução inviável aplicando um procedimento de reparo para garantir a viabilidade.

O primeiro algoritmo construtivo segue a mesma lógica de construção do conjunto de ofertas proposta em (Nobibon *et al.*, 2011), baseada em $NPP_{ij} = (pij - cij)/cij$, com um procedimento adicional para inclusão de produtos e a capacidade de lidar com o canibalismo. Com a regra de priorizar ofertas de maior lucro, existe a possibilidade de conjuntos de ofertas viáveis serem desconsiderados, especialmente quando o orçamento está apertado. Por exemplo, no caso em que um produto tem um orçamento apertado e a maioria de suas ofertas mais lucrativas tem um custo alto, o conjunto de ofertas resultante atingirá o orçamento máximo antes de atingir o número mínimo de ofertas necessárias para o produto. Para resolver essa limitação, uma etapa adicional é incluída no ponto em que o conjunto de

ofertas de produtos é concluído pelo procedimento original. Para cada produto não incluído na solução, essa etapa adicional tenta criar um novo conjunto de ofertas, com base em c_{ij} , em vez de NPP_{ij} . Se um conjunto de ofertas viável for encontrado com um lucro maior do que o original, será substituído. Para o segundo algoritmo construtivo, utilizamos uma versão aleatória gananciosa do primeiro algoritmo, ou seja, em vez de escolher ofertas em ordem decrescente de lucratividade, ele selecionará uma oferta aleatória entre os 10% melhores.

O terceiro algoritmo é um algoritmo determinístico que aborda o processo de construção da solução de um ângulo diferente, ou seja, começando com uma solução inviável para convertê-la em uma solução viável aplicando um procedimento de reparo. Essa abordagem começa com um conjunto de ofertas para cada produto que extrapola o orçamento de cada produto, ignorando quaisquer restrições de problema. Depois que cada conjunto de ofertas é produzido, o algoritmo tenta reparar cada conjunto de ofertas de produtos na viabilidade, indo em ordem decrescente de lucratividade dos conjuntos de ofertas inviáveis. Damos maior importância à parte viável do conjunto de ofertas, contabilizando todo o lucro proporcionado pelas ofertas viáveis e apenas 20% do lucro obtido pelas ofertas inviáveis. Para reparar a solução em uma solução viável, esse processo tenta gerenciar conflitos de oferta causados por mais produtos oferecidos a um cliente do que seu limite de oferta. Esse reparo será feito substituindo a oferta conflitante em um dos produtos de forma que nenhuma outra restrição de problema seja violada.

As soluções produzidas pelos três algoritmos construtivos discutidos anteriormente são usadas como soluções iniciais para o Algoritmo de Busca Tabu. Eles também são a base da população inicial do Algoritmo Genético.

3.2 TABU PESQUISAR

O algoritmo Tabu Search (Glover, 1989) proposto aqui conduz um método de pesquisa local com três bairros que são explorados sequencialmente. As vizinhanças 1 e 2, digamos, $N1(x,y)$ e $N2(x,y)$, foram propostas por (Nobibon et al., 2011) e podem ser formalizadas da seguinte forma: O conjunto $N1(x,y)$ contém as soluções viáveis (x', y') obtidas de (x, y) considerando duas clientes i e h , e um produto j satisfazendo $yj = 1; xij = 1$ e $xhj = 0$; então definimos $x'_{ij} = 0$ e $x'_{hj} = 1$. O conjunto $N2(x,y)$ contém as soluções viáveis (x', y') obtidas de (x, y) considerando dois clientes i e h , e dois produtos j e l satisfazendo $yj = yl = 1; xij = 1; xhj = 0; xil = 0$ e $xhl = 1$; então definimos $x'_{ij} = 0; x'_{hj} = 1; x'_{il} = 1$ e $x'_{hl} = 0$.

A vizinhança 3, $N3(x,y)$, descrita no Algoritmo 6, tenta combinar as melhores características de $N1(x,y)$ e $N2(x,y)$, e de forma a permitir mudanças no número de clientes alocados a cada produto

ativo, ou seja, a cardinalidade do conjunto de clientes alocados a cada produto. No final, ele tenta inserir novas ofertas na solução final. O movimento de swap em $N3(x,y)$, é representado por uma tupla (j, l, i) e significa que um cliente i pode ser movido do produto j para l . Inicialmente, ele explorará cada cliente classificado em ordem não crescente de sua variação de lucro, tentando encontrar trocas únicas lucrativas entre produtos ativos e armazenando até duas trocas inviáveis mais lucrativas. Cada vez que uma única troca é encontrada, é realizada uma busca por um movimento de intercâmbio com outro cliente que viabilize toda a operação. Se algum movimento de intercâmbio for encontrado, ele será feito imediatamente. Depois que todos os clientes foram examinados, há um teste final para inserir ofertas lucrativas nos produtos ativos. Até onde sabemos, esta é a primeira vizinhança que altera efetivamente o número de clientes oferecidos por cada produto, e esse recurso permite que o método proposto lide com todas as classes de instâncias propostas, conforme mostrado na seção 4.

O algoritmo Tabu Search (TS) (ver Algoritmo 1) proposto aqui percorre de forma inovadora os bairros superando os problemas de taxa de convergência encontrados em (Nobibon *et al.*, 2011), principalmente em grandes instâncias, onde cada movimento causa um impacto muito baixo em termos da função objetivo. A primeira melhoria é executar qualquer movimento lucrativo assim que for encontrado. O tamanho dos bairros também foi controlado para $N1$ e $N2$, no início, apenas 70% dos clientes ordenados em ordem não crescente de seu $NPPij$ são considerados para análise e quando não se encontram movimentações mais lucrativas esse percentual cresce para permitir mais combinações. Aqui foram aplicados os princípios da Pesquisa de Vizinhança Variável (VNS) (Hansen, P. e Mladenović, 2001). Essas modificações, juntamente com o Algoritmo Genético embarcado, tornaram o algoritmo proposto mais rápido e alcançando soluções melhores do que a literatura.

Algoritmo 1. Procedimento de pesquisa tabu

```

1: Iter := 0, tabu_list := { }, α := 0, escolha uma solução inicial  $(x, y)$ ,  $x^* := x' := x$ ,  $y^* := y' := y$ 
2: quando ( $Iter < 30$  e tempo  $< 300$  seg) fazer
   3:  $(x_{prev}, y_{prev}) := (x', y')$ 
   4:  $(xn, yn) := explore\ N1(x', y')$ , considerando tabu_list
5: se  $de(xn, yn) > de(x', y')$ , onde  $de(x, y)$  é o valor da função objetivo da solução  $(x, y)$  então
   6:            $x' := xn$ ,  $y' = yn$ , atualização tabu_list
   7: endif
   8:  $(xn, yn) := explore\ N2(x', y')$ , considerando tabu_list
   9: se  $de(xn, yn) > de(x', y')$  então
  10:            $x' := xn$ ,  $y' = yn$ , atualizar tabu_list
  11: endif
  12:  $(xn, yn) := explore\ N3(x', y')$ , considerando tabu_list
  13: se  $de(xn, yn) > de(x', y')$  então
  14:            $x' := xn$ ,  $y' = yn$ , atualização tabu_list
  15: endif
  16: se  $de(x_{prev}, y_{prev}) \leq de(x', y')$  então
  17:           Tijolo := Tijolo + 1
  18: se ( $Iter \bmod 10$ ) == 0

```

```

19:           Atualizar a população do GA
20:           Expanda o tamanho da vizinhança TS
21:         endif
22: se Iter > 1 então
23:            $\alpha := \alpha + 0.03$ 
24: ( $x'$ ,  $y'$ ) := executa a Regressão Genética para ( $x'$ ,  $y'$ ) e  $\alpha$ 
25: senão
26: ( $x'$ ,  $y'$ ) := executa a otimização genética para ( $x'$ ,  $y'$ )
27: endif
28: endif
29: se  $de(x^*, y^*) < de(x', y')$  então
30:    $x^* := x'$ ,  $y^* = y'$ 
31:   Iter := 0,  $\alpha := 0.0$ 
32:           Redefinir o tamanho da vizinhança TS
33: endif
34: enquanto

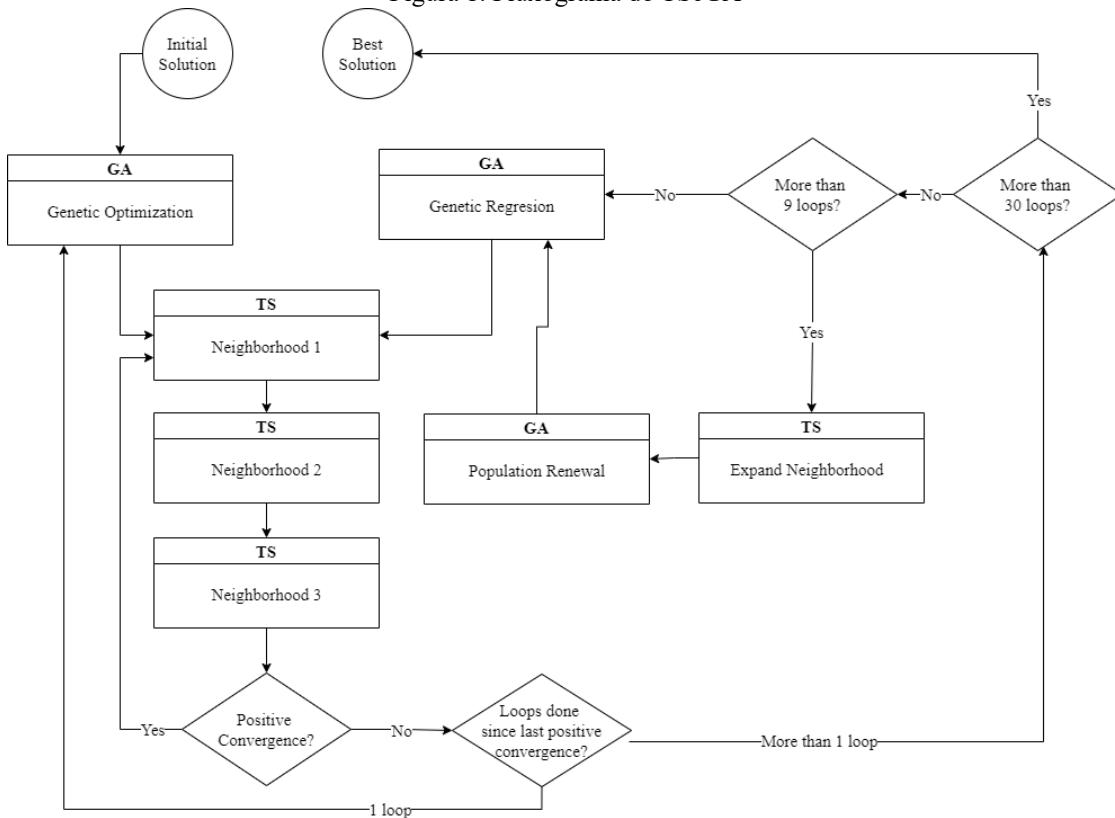
```

Como o TS pode realizar vários movimentos em cada iteração, a lista tabu é adaptada da seguinte maneira. O atributo de movimento é a tupla (j, l, eu) , e o que significa que um cliente eu pode ser movido do produto j . Para l , lembre-se, que para os bairros 1 e 2, $j = l$. Então, sempre que um movimento de não melhoria é executado, o movimento de reversão é proibido apenas por 2 iterações, e esse foi o melhor para comprometer a qualidade da solução e o esforço computacional. Considerando que o TS explora exaustivamente todas as vizinhanças em cada iteração, e cada movimento geralmente causa pequenos aumentos no valor da função objetivo, outra inovação foi o uso do Algoritmo Genético como um conjunto de possíveis formas alternativas de continuar a busca quando o TS não consegue encontrar movimentos melhorados por duas iterações consecutivas. Neste ponto, o TS solicitará a melhor solução na população atual do GA com uma pontuação de aptidão de pelo menos α (começando em 3%) por cento do valor da função objetivo atual. Quando uma nova solução existente é encontrada ou uma nova população é gerada, o valor α é redefinido, caso contrário, é aumentado em 3%.

3.3 ALGORITMO GENÉTICO

O GA é detalhado no Algoritmo 2, onde o processo de reprodução é aplicado à população gerando um novo que substitui inteiramente a população antiga, mas mantendo o indivíduo mais bem avaliado (Elite) na próxima geração. Cada novo indivíduo é criado pelo cruzamento (Algoritmo 9) entre dois pais selecionados da população anterior. Os pais são selecionados ao longo de uma roleta ponderada com base na aptidão (nesse caso, o valor da função objetivo). Esta implementação do GA utiliza dois tipos de procedimentos de mutação. O primeiro só estará ativo em casos problemáticos com canibalismo e será aplicado a cada indivíduo gerado pior do que a melhor solução até agora, trocando o pai dominante pelo pai de menor aptidão. Isso mudará a ordem de bloqueio de produtos canibais, produzindo descendentes diferentes.

Figura 1. Fluxograma do TSeGA



Nenhuma memória de longo prazo foi usada na ST, mas como a população em GA converge rapidamente para soluções muito semelhantes à melhor solução de TS, um procedimento de diversificação foi implementado em GA, como o segundo tipo de mutação, que tem 75% de chance de ser aplicada para cada indivíduo gerado que tenha menor aptidão do que seus pais. Quando são feitas comparações de similaridade entre indivíduos da população GA, é fácil obter quais ofertas são feitas com mais frequência. Então, no final de cada geração, uma lista de ofertas (tamanho máximo de $0,1n$) pertence a mais de 55% dos indivíduos da população GA. A Mutação usa esta lista para proibir que essas ofertas sejam selecionadas, a escolha é feita aleatoriamente e, enquanto seu número não ultrapassa a metade do número de elementos da lista. Todo o processo de interação entre GA e ST é mostrado na Figura 1. Detalhes da estrutura cromossômica, cruzamento e mutação podem ser encontrados (Müller et al. 2018).

Algoritmo 2. Procedimento de Algoritmo Genético

```

1: Inicialize a população GA executando 200 vezes o algoritmo 3: Algoritmo construtivo aleatório ganancioso
2:  $BS :=$  indivíduo com maior aptidão na população
3:  $Geração := 0$ ,  $Offer\_List := \{\}$ 
4: enquanto ( $Geração < 4$ )
5:    $New\_Population := \{\}$ 
6:    $Elite := BS$ 
  
```

```

    7: enquanto ( $|New\_Population| < 199$ )
8:   Pai1           := fitness_weighted_roulette(População)
9:   Parent2 := fitness_weighted_roulette(População \ {Parent1})
10:  Dominante := indivíduo com maior aptidão entre Parent1 e Parent2
11:  Pai2 := indivíduo com menor aptidão entre Pai1 e Pai2
12: descendência := aplique o Algoritmo 9 (Crossover) para Dominante e Pai2
    13: se (aptidão (prole) < aptidão (Elite))
    14: se  $\exists (k,j) / k \in Pai1 \text{ e } j \in Pai2 \text{ e } (k,j) \in Pode$ 
        Dominante := indivíduo com menor aptidão entre Parent1 e Parent2
        Pai2 := indivíduo com maior aptidão entre Pai1 e Pai2
15:  16:   descendência := aplique o Algoritmo 9 (Crossover) para Dominante e Pai2
    17:   18: endif
        19: endif
    20: se (aptidão (descendência) < aptidão (Elite) e Geração > 2)
        21: se (aleatório (0,1) <= 0,75)
        22: Aplique Mutação à prole usando Offer_List
            23: endif
            24: endif
    25:   New_Population := New_Population  $\cup \{descendência\}$ 
    26: se (aptidão (descendência) > aptidão (Elite))
        27:   Elite := descendência
        28: endif
        29: Fim
    30:   População = New_Population  $\cup \{Elite\}$ 
    31: se (fitness(Elite) > fitness(BS))
        32: BS := Elite
        33: endif
    34:   Offer_List := construir a lista de similaridade a partir de New_Population
    35:   Geração := Geração + 1
    36: terminar

```

4 EXPERIMENTOS COMPUTACIONAIS

Implementamos os dois procedimentos heurísticos propostos por (Cetin e Alabas-Uslu, 2015) e TSeGA comparando com os resultados apresentados por (Nobibon et al., 2011) e as soluções exatas ou limites superiores obtidos por solucionadores comerciais (GUROBI e CPLEX). As primeiras instâncias usadas para testar foram disponibilizadas por (Nobibon et al., 2011) denominadas S1, S2, S3, M1, M2 e, L para cada combinação de $m = 100, 200, 300, 1000, 2000, 10000$; $n = 5, 10, 15$; e $R = 5\%, 10\%, 15\%$; três maneiras aleatórias diferentes de gerar o orçamento Bj e duas maneiras diferentes de gerar Mi , compreendendo 324 instâncias denominadas Grupo 1. Para cada um desses casos, geramos um conjunto com $(n/5)$ par de produtos canibais por meio de uma análise estatística identificando o mais semelhante pela Distância Euclidiana.

Para comprovar a robustez do TSeGA, também geramos um conjunto de instâncias muito grandes, a primeira, denominada XL, segue o método de geração de (Nobibon et al., 2011), todas usando o valor aleatório intermediário de Bj e o Mi mais apertado com: $m = 15000, 20000$; $n = 10, 20, 30, 40, 50$; $R = 10\%$ (10 instâncias); $m = 40000$; $n = 5, 10, 15, 40$; $R = 10\%$ (4 instâncias); $m = 50000$; $n = 15$; $R = 15\%$ (1 instância); e $m = 100000$; $n = 15$; $R = 15\%$ (1 instância) totalizando 16

instâncias denominadas Grupo 2. Para cada uma dessas instâncias, geramos um conjunto com ($n/5$) par de produtos canibais por meio de uma análise estatística identificando o mais semelhante pela Distância Euclidiana e também por um método denominado Similaridade e Dissimilaridade. Similaridade e Dissimilaridade fazem uso da melhor solução encontrada para cada instância, então os pares canibais são definidos como o par de produtos oferecidos para o maior número de clientes idênticos (Similaridade) ou clientes diferentes (Dissimilaridade). Os resultados serão apresentados para cada grupo de instâncias para melhor visualizar a superioridade e robustez do TSeGA. A Tabela 1 resume os recursos computacionais para implementar os algoritmos.

Tabela 1. Resumo dos Recursos de Implementação

Algoritmo	Referências	Recursos Computacionais
Exacto	Schneider, 2019 e Cruz, 2020	IBM ILOG CPLEX e ZIMPL
HR-1	Cetin e Alabas-Uslu, 2015, com regra heurística 1	Gurobi 8.1.1 com Julia Pro 1.2.0-1 e JuMP
HR-2	Cetin e Alabas-Uslu, 2015, com regra heurística 2	Gurobi 8.1.1 com Julia Pro 1.2.0-1 e JuMP
TSeGA		Comunidade do Visual Studio usando C++ (Visual Studio, 2019)

Primeiro, definimos como limite superior (UB) o valor ótimo da função objetivo ou o melhor limite encontrado pelo solucionador e como limite inferior (LB) o valor da função objetivo obtido pelo método heurístico. A métrica utilizada para comparar e avaliar a qualidade dos métodos foi o gap percentual definido como $\Delta = ((UB - LB)/UB).100\%$, se alguma instância não for resolvida pelo método heurístico consideramos $\Delta = 100\%$.

4.1 INSTÂNCIAS DO GRUPO 1

Iniciamos a análise considerando 324 instâncias originalmente apresentadas por (Nobibon et al., 2011) e também os três tipos de instâncias, incluindo o canibalismo. A Tabela 2 mostra a frequência de cada algoritmo na obtenção da melhor solução. Se olharmos apenas para a Tabela 2 podemos observar que a HR-1 tem um desempenho semelhante ao da TSeGA, observarmos a Tabela 3 mais adiante com os gaps percentuais, destaca-se a superioridade da TSeGA sobre a HR-1 e HR-2 mesmo quando tomamos a média entre a melhor solução encontrada pela HR-1 e a HR-2.

Tabela 2. Número de melhores soluções encontradas pelos algoritmos GATeS, HR-1 e HR-2 - Grupo 1

	Número de melhores resultados encontrados
	Canibalismo

Algoritmo heurístico	Problema original	Distância Euclidiana (ED)	Similaridade	Dissimilaridade
TSeGA	108	129	128	100
HR-1	109	75	80	75
HR-2	39	46	55	79
TSeGA = HR-1	3	3	5	4
TSeGA = HR-2	6	5	5	7
TSeGA = HR-1 = HR-2	9	9	10	10
HR-1 = HR-2	50	57	41	49

Tabela 3 – Gaps médios de TSeGA, HR-1 e HR-2 – Instâncias do Grupo 1

Problema	GAPs médios ($\Delta\%$)			
	TSeGA	HR-1	HR-2	Melhor entre HR-1 e HR-2
Problema original	2.09	6.69	4.35	2.12
Canibalismo - ED	1.75	7.20	5.13	2.72
Canibalismo - Semelhança	1.73	6.79	5.31	2.95
Canibalismo - Dissimilaridade	1.97	7.15	3.71	2.08

A Tabela 4 mostra a falta de robustez do HR-1 e do HR-2, uma vez que eles não são capazes de resolver alguns casos sem canibalismo e esse comportamento se intensifica quando incluímos o canibalismo.

Tabela 4. Número de instâncias não resolvidas por TSeGA, HR-1 e HR-2

Algoritmo heurístico	Número de instâncias não resolvidas			
	Problema original	Canibalismo		
		Distância Euclidiana (ED)	Similaridade	Dissimilaridade
TSeGA	0	0	0	0
HR-1	3	1	1	1
HR-2	3	9	9	9

A seguir, apresentaremos os resultados para as instâncias do Grupo 2 que são maiores e mais difíceis de resolver do que as instâncias do Grupo 1.

4.2 INSTÂNCIAS DO GRUPO 2

Os resultados para o Grupo 2 estão resumidos na Tabela 5, onde a TSeGA mostra um desempenho quase metade do valor dos gaps médios de HR-1, HR-2 e o melhor de ambos.

Tabela 5. Gaps médios de TSeGA, HR-1 e HR-2 – Instâncias do Grupo 2

Problema	GAPs médios ($\Delta\%$)			
	TSeGA	HR-1	HR-2	Melhor entre HR-1 e HR-2
Problema original	3.47	7.91	6.81	6.59
Canibalismo - ED	4.98	15.41	15.60	14.90
Canibalismo - Semelhança	7.30	14.54	19.60	14.52
Canibalismo - Dissimilaridade	6.45	13.13	16.83	12.80

Como podemos ver, o TSeGA tem um desempenho muito melhor do que o HR-1 e o HR-2, apresentando um Gap sempre melhor do que o HR-1 e o HR-2 e, para instâncias do Grupo 1, o Gap é quase metade do melhor Gap entre HR-1 e HR-2. Não apresentamos os resultados de (Nobibon et al., 2011) porque HR-1 e HR-2 foram sempre superiores, e também retemos os tempos computacionais porque as corridas foram realizadas em máquinas diferentes e TSeGA como um limite de tempo de 600 segundos por ciclo.

5 CONCLUSÕES

Este artigo apresentou uma heurística híbrida, denominada TSeGA. É uma pesquisa tabu com um algoritmo genético incorporado. Os resultados computacionais mostram a qualidade do TSeGA obtendo melhores resultados do que outras heurísticas para grande margem quando consideramos as lacunas percentuais. A inclusão da restrição de canibalismo tornou o problema mais difícil para os outros métodos, enquanto o TSeGA poderia resolvê-lo sem sacrificar seu desempenho.

Também mostramos que HR-1 e HR-2 não foram robustos, pois há casos em que eles não foram capazes de resolver, e essa limitação não foi descoberta antes na literatura.

Por fim, podemos concluir que o TSeGA é um método promissor para resolver o problema do marketing direto, incluindo ou não a restrição de canibalismo, e mesmo que necessite de algum esforço computacional extra, vale a pena, pois esse problema é um planejamento de longo prazo.

CONFIRMAÇÕES

Este trabalho foi parcialmente apoiado pela CAPES e pelo programa de pesquisa e inovação Horizonte 2020 da União Europeia sob o acordo de subvenção nº 739551 (KIOS CoE) e pelo Governo da República de Chipre através da Direção Geral de Programas, Coordenação e Desenvolvimento Europeus.

REFERÊNCIAS

- ABEDI, V. S. Allocation of advertising budget between multiple channels to support sales in multiple markets. *Journal of the Operational Research Society*, 68(2), 134–146, 2017.
- BHASHAR, T.; SUNDARARAJAN, R.; KRISHNAN, P. G. A fuzzy mathematical programming approach for cross-sell optimization in retail banking. *Journal of the Operational Research Society*, 60(5), 717–727, 2009.
- CETIN, F.; ALABAS-USLU, C. Heuristic solution to the product targeting problem based on mathematical programming. *International Journal of Production Research*, 55(1), 3–17, 2015.
- COELHO, V. N. et al. Generic Pareto local search metaheuristic for optimization of targeted offers in a bi-objective direct marketing campaign. *Computers and Operation Research*, 70, 578–587, 2017. Disponível em: <http://dx.doi.org/10.1016/j.cor.2016.09.008>.
- COHEN, M. D. Exploiting response models: optimizing cross-sell and up-sell opportunities in banking. *Information Systems*, 29(4), 327–341, 2004.
- CRABBE, M.; LIEBERMAN, G.; MORIARTY, S. Consumer Trends 2030. Disponível em: <https://www.mintel.com>, 2019.
- CZIKOSOVA, A.; ANTOŠOVA, M.; ČUKOLOVA, K. Strategy in direct and interactive marketing communications. *Procedia - Social and Behavioral Sciences*, 116, 1615–1619, 2014.
- GLOVER, F. Tabu search: part I. *ORSA Journal on Computing*, 1(3), 190–206, 1989.
- HANSEN, P.; MLADENOVIĆ, N. Variable neighborhood search: principles and applications. *European Journal of Operational Research*, 130(1), 449–467, 2001.
- KOTLER, P.; ARMSTRONG, G. *Principles of Marketing* (16. ed.). Pearson, New Jersey, 2016.
- ŁADYZYNSKI, P.; ZBIKOWSKI, K.; GAWRYSIAK, P. Direct marketing campaigns in retail banking with the use of deep learning and random forests. *Expert Systems With Applications*, 134, 28–35, 2019.
- MÜLLER, F. M.; BONILHA, I. S.; SOUZA, V. B. A hybrid approach (genetic algorithm + tabu search) to solve the direct marketing problem. In: *Proceedings of XIX CLAIO*, Lima, Peru, 248–255, 2018.
- NOBIBON, F. T.; LEUS, R.; SPIEKSM, F. C. Optimization models for targeted offers in direct marketing: exact and heuristic algorithms. *European Journal of Operational Research*, 210(3), 670–683, 2011.
- OLIVEIRA, T. A. et al. A hybrid variable neighborhood search algorithm for targeted offers in direct marketing. *Electronic Notes in Discrete Mathematics*, 47, 205–212, 2015.
- PRAAG, N. V. Optimization of promotion campaigns using tabu search. 2010. Master Dissertation. Faculty of Business and Economics, KULeuven, Belgium.

SCHNEIDER, V. A. Análise de modelos matemáticos para o problema de marketing direto com oferta de produtos. 2019. Master Dissertation. PPGEP/UFSM, Brazil.

SOUZA, V. B. Uma abordagem meta-heurística para o problema de marketing direto considerando o canibalismo entre produtos. 2018. Master Dissertation. PPGA/UFSM, Brazil.