UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

CENTRO DE CIÊNCIAS AGRÁRIAS

DEPARTAMENTO DE COMPUTAÇÃO

LEANDRO PETERLE VANELI

CLUSTERING SEARCH PARA RESOLUÇÃO DE UM PROBLEMA DE PROGRAMAÇÃO DE ROTEIROS TURÍSTICOS

ALEGRE - ES

2015

LEANDRO PETERLE VANELI

CLUSTERING SEARCH PARA RESOLUÇÃO DE UM PROBLEMA DE PROGRAMAÇÃO DE ROTEIROS TURÍSTICOS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.
Orientador: Prof. Dr. Geraldo Regis Mauri.

ALEGRE - ES
2015

LEANDRO PETERLE VANELI

CLUSTERING SEARCH PARA RESOLUÇÃO DE UM PROBLEMA DE PROGRAMAÇÃO DE ROTEIROS TURÍSTICOS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Aprovado em 25 de novembro de 2015.

COMISSÃO EXAMIDADORA

Prof. Dr. Geraldo Regis Mauri
Universidade Federal do Espírito Santo
Orientador

Prof. M.Sc. Rodrigo Freitas Silva
Universidade Federal do Espírito Santo

Prof. M.Sc. Edmar Hell Kampke
Universidade Federal do Espírito Santo

“É preciso diminuir a distância entre o que se diz e o que se faz, até que, em um dado momento, a tua fala seja a tua prática.”

PAULO FREIRE

AGRADECIMENTOS

A Deus por ter me dado forças quando eu mais precisei e me fazer sempre ter foco na busca dos meus sonhos.

A minha família, principalmente a meu pai Gilson Luiz Vaneli, minha mãe Nilzete Maria Peterle Vaneli e meu irmão Bruno Peterle Vaneli que não mediram esforços nenhum para me ajudar a alcançar mais uma conquista.

Aos meus amigos que foram essenciais ao longo dessa jornada, principalmente a Ana Paula Borges Simiqueli, André Soares que sempre estiveram ao meu lado dando forças para nunca desistir e, ao Sávio Cunha de Souza que foi meu parceiro e companheiro de casa estando sempre ao meu lado nas horas boas e ruins.

Ao professor Geraldo Regis Mauri, pela paciência e orientação desde o projeto de iniciação científica até esse trabalho de conclusão de curso.

A esta universidade, sua direção, administração e principalmente ao departamento de computação, com seus excelentes profissionais no qual puderam me passar um pouco de seu conhecimento.

A empresa júnior Click Júnior por me proporcionar uma pequena experiência do mercado de trabalho aumentando assim o conhecimento adquirido nas aulas.

A todos, muito obrigado.

RESUMO

Este trabalho tem como objetivo apresentar o Problema de Programação de Roteiros Turísticos (PPRT). O PPRT consiste em criar roteiros turísticos dentro de um conjunto de locais para visitação, levando em consideração algumas restrições como o dinheiro máximo que o turista tem disponível, os horários de abertura e fechamento de cada local, entre outras. Para resolver o PPRT é proposta a utilização da meta-heurística híbrida *Clustering Search* (CS), utilizando o método *Simulated Annealing* (SA) como gerador de soluções. A CS foi aplicada a instâncias de diferentes tamanhos, geradas a partir de dados reais referentes ao estado do Espírito Santo. O desempenho da CS foi avaliado por meio de experimentos computacionais considerando diferentes situações, com e sem restrição de dinheiro, obtendo bons resultados em todos os casos.

Palavras-chave: *Clustering Search*; *Simulated Annealing*; Problema de Programação de Roteiros Turísticos; Meta-heurística Híbrida.

CLUSTERING SEARCH TO SOLVE A TOURIST TRIP DESIGN PROBLEM

ABSTRACT

This work aims to introduce the Tourist Trip Design Problem (TTDP). The TTDP consists of creating tours considering a set of locations to visit. Some constraints such as available money and opening and closing times of each location must be considered. The hybrid metaheuristic Clustering Search (CS) was applied to solve the TTDP by using a Simulated Annealing (SA) to generate solutions. The CS was applied to instances of different sizes. These instances were generated from real data regarding the state of Espírito Santo. The performance of the CS was evaluated through computational experiments considering different situations, with and without money constraint, and good results were obtained in all cases.

Keywords: Clustering Search; Simulated Annealing; Tourist Trip Design Problem; Hybrid Metaheuristic.

LISTA DE FIGURAS

Figura 1 - Cachoeira da Fumaça, Alegre - ES.....12

Figura 2 - Convento da Penha, Vila Velha - ES. 12

Figura 3 - Imagem do percurso rodoviário entre a Cachoeira da Fumaça e o Convento da Penha.....13

Figura 4 - Exemplo de Instância com 5 locais e 2 dias para visitação. 20

Figura 5 - Fluxograma da CS 21

Figura 6 - Pseudocódigo da SA.....23

Figura 7 - Representação de uma solução para o PPRT. 23

Figura 8 - Exemplo de uma Rota no *Google Maps* 24

Figura 9 - Exemplo de cálculo da distância de Hamming.....25

Figura 10 - Pseudocódigo da CS implementada. 28

LISTA DE TABELAS

Tabela 1 - Calibração da SA. 29

Tabela 2 - Calibração da CS. 30

Tabela 3 - Desempate da Calibração da CS. 31

Tabela 4 - Melhores parâmetros para a SA e a CS.....31

Tabela 5 - Melhores soluções SEM a restrição de dinheiro. 32

Tabela 6 - Melhores soluções COM a restrição de dinheiro.....32

Tabela 7 - Visão geral das execuções SEM a restrição de dinheiro. 33

Tabela 8 - Visão geral das execuções COM a restrição de dinheiro.....34

Tabela 9 - Pontos turísticos.....38

LISTA DE ABREVIATURAS E SIGLAS

CS – *Clustering Search*

ECS – *Evolutionary Clustering Search*

FO – Função Objetivo

GRASP – *Greedy Randomized Adaptive Search Procedure*

MGT – *Mobile Tourist Guide*

OC – Otimização Combinatória

OP – *Orienteering Problem*

PERT – Problema de Elaboração de Rotas Turísticas

PPRT – Problema de Programação de Roteiros Turísticos

SA – *Simulated Annealing*

TOP – *Team Orienteering Problem*

TTDP – *Tourist Trip Design Problem*

SUMÁRIO

1. INTRODUÇÃO 12

1.1. O PROBLEMA E SUA IMPORTÂNCIA 12

1.2. OBJETIVOS 14

1.2.1. Objetivo geral 14

1.2.2. Objetivos específicos 14

1.3. ESTRUTURA DO TRABALHO 15

2. REVISÃO DA LITERATURA 16

3. METODOLOGIA 18

3.1. COLETA DOS DADOS 18

3.2. GERAÇÃO DE INSTÂNCIAS 18

3.3. CLUSTERING SEARCH 21

3.3.1. Principais Componentes da CS 22

3.3.1.1. Gerador de soluções 22

3.3.1.2. Agrupamento iterativo 25

3.3.1.3. Analisador de agrupamentos 26

3.3.1.4. Busca local 26

3.3.1.5. Função objetivo 26

3.3.2. Algoritmo CS 27

4. RESULTADOS 29

5. CONCLUSÕES 35

REFERÊNCIAS BIBLIOGRÁFICAS 36

APÊNDICE A. 38

1. INTRODUÇÃO

Problemas de Otimização Combinatória (OC) têm como objetivo em sua maioria maximizar ou minimizar funções de um certo domínio específico. Geralmente, em problemas de OC, esse domínio é finito, e então um elemento desse domínio pode ser testado. Porém, testar todos os possíveis elementos a procura do melhor seria uma tarefa considerada inviável, pois mesmo que a instância em questão não seja tão grande assim, encontrar tal elemento demandaria muito tempo. Tais problemas estão presentes em várias áreas conhecidas, e até mesmo no nosso dia-a-dia, como por exemplo em cortes de chapas de aço, roteamento de veículos, rotação de culturas, alocação de grades de horários, entre outros. Problemas de OC são, em sua maioria, difíceis de encontrar a solução ótima, devido a sua natureza combinatória e, em alguns casos, encontrar soluções viáveis torna-se uma tarefa exaustiva e até mesmo impraticável.

1.1. O PROBLEMA E SUA IMPORTÂNCIA

O turismo no Brasil vem crescendo cada vez mais, pois seu grande território abriga uma quantidade enorme de pontos turísticos dos mais variados tipos. Pessoas do mundo todo estão sendo atraídas por sua beleza única, buscando pontos turísticos novos que os surpreendam com suas belezas e seus encantos.

Um de seus estados, o Espírito Santo, contém diversos pontos turísticos, sendo um desses a Cachoeira da Fumaça, localizada na cidade de Alegre e ilustrado na Figura 1. Um outro local que é muito procurado pelos turistas é o Convento da Penha, que fica localizado na cidade de Vila Velha, e é ilustrado na Figura 2.

Figura 1 - Cachoeira da Fumaça, Alegre - ES.

Figura 2 - Convento da Penha, Vila Velha - ES.

Esses dois locais se encontram geograficamente distantes um do outro. A distância rodoviária entre eles é de 204 km. Sendo assim, para um turista ir de carro da Cachoeira da Fumaça para o Convento da Penha, ele gastaria aproximadamente 3 horas e 31 minutos de viagem, como mostra a Figura 3.

Figura 3 - Imagem do percurso rodoviário entre a Cachoeira da Fumaça e o Convento da Penha
Fonte: *Google Maps*[1].

Para o turista considerar esses dois pontos de viagem em um mesmo dia ficaria complicado, pois o tempo de viagem é longo. Com isso, para que esses dois locais estivessem no mesmo roteiro de viagem de um determinado dia, a satisfação desse turista em relação a estes locais tem que ser muito alta, ou seja, a vontade de visita-los. Além disso, também há o motivo do Convento da Penha ficar em uma cidade onde há outros pontos turísticos por perto, o que permite que o turista visite mais locais e tenha uma melhor "satisfação", tornando assim a viagem longa mais viável.

Os pontos turísticos do estado do Espírito Santo variam bastante, tendo um leque grande de locais para visitação de diferentes tipos, como praias, cachoeiras, rapel, trilhas, voos de parapentes, lindas paisagens, artesanatos, entre muitos outros tipos de turismo que se queira fazer.

[1] Acessado em https://goo.gl/Hm9Qsg. Podendo haver modificação no tempo gasto de viagem dependendo do dia e hora acessada.

Entretanto, com tantos locais para se visitar, fica difícil para o próprio turista planejar uma boa rota que atenda suas necessidades da melhor forma possível, pois o mesmo geralmente não tem dinheiro e tempo disponível para visitar todos os locais de seu interesse. Locais como os apresentados nas Figuras 1 e 2 são muito distantes um do outro, dificultando ainda mais o planejamento da rota turística. Em geral, os turistas possuem alguns dias disponíveis para visitação, e uma quantidade máxima de dinheiro que se pretende gastar em toda a programação de visitações. Já os locais que serão visitados têm horário de abertura e de fechamento em sua grande maioria, como por exemplo, o Convento da Penha. Sendo assim, essa informação deve ser considerada também, pois não se pode visitar um local que não esteja aberto para visitação. Além disso, para cada local visitado existe um tempo mínimo que o turista deve permanecer no local, para que possa conhecer o local de forma “adequada”. Para um turista comum planejar e considerar tudo isso sozinho, sem ajuda de alguma ferramenta, fica praticamente impossível.

Assim, neste trabalho, busca-se implementar uma meta-heurística para resolver tal problema, considerando todas as variáveis já mencionadas com o intuito de auxiliar o turismo no estado do Espírito Santo.

1.2. OBJETIVOS

A seguir são apresentados os objetivos gerais e específicos relacionados a este trabalho.

1.2.1. Objetivo geral

O objetivo geral deste trabalho é a aplicação da meta-heurística *Clustering Search* (CS) com a heurística geradora de solução *Simulated Annealing* (SA) para resolução de um Problema de Programação de Roteiros Turísticos (PPRT) utilizando dados reais de pontos turísticos do estado do Espírito Santo.

1.2.2. Objetivos específicos

a) Desenvolver as adaptações necessárias para modelagem do PPRT;
b) Implementar a CS utilizando a meta-heurística SA como geradora de solução;
c) Obter dados reais por meio do *Google Maps* com diferentes pontos turísticos do estado do Espírito Santo para geração de diferentes instâncias;

d) Avaliar o desempenho do método por meio de experimentos computacionais.

1.3. ESTRUTURA DO TRABALHO

O restante desse trabalho está dividido da seguinte forma. O Capítulo 2 apresenta a revisão da literatura, contendo os principais trabalhos relacionados a este problema. A metodologia deste trabalho é descrita no Capítulo 3, no qual são apresentados a coleta dos dados, a CS e seus principais componentes. No Capítulo 4 são descritos os resultados obtidos, e as conclusões são apresentadas no Capítulo 5.

2. REVISÃO DA LITERATURA

É possível encontrar na literatura alguns trabalhos relacionados ao Problema de Programação de Roteiros Turísticos (PPRT) e sua aplicabilidade em diferentes abordagens. A seguir, é apresentada uma breve revisão bibliográfica e as abordagens utilizadas em alguns dos principais trabalhos relacionados a esse problema. O *Team Orienteering Problem* (TOP), que é um tipo de *Orienteering Problem* (OP), é a base para o PPRT. Tais problemas serão apresentados no decorrer desse capítulo.

Um problema de *Orienteering Problem* (OP) consiste em criar uma rota maximizando a pontuação dos locais visitados. Os locais de saída e chegada são fixos, e existe um tempo de percurso entre cada local e um tempo total para percorrer tal percurso (*Tmax*). Golden et al. (1987) provaram que o OP é NP-difícil e, como consequência, o TOP também é NP-difícil.

Chao et al. (1996) foram um dos pioneiros nessa área. Os autores apresentaram uma nova heurística para resolução do *Team Orienteering Problem* (TOP). Nesse problema, uma equipe tenta maximizar a pontuação de locais visitados, ou seja, é definido um subconjunto de locais para cada integrante da equipe sem repetição de locais, ou seja, um local não pode ser visitado por mais de um integrante da equipe. Existe uma limitação de tempo para que a equipe faça o percurso e, ao final do tempo, é exibida a pontuação da equipe. O objetivo nesse caso é maximizar essa pontuação.

Vansteenwegen e Van Oudheusden (2007) apresentaram uma ideia para solucionar problemas de roteiros turísticos em geral, pois os turistas geralmente utilizam *sites*, artigos de revistas ou guias para criar seus roteiros turísticos. O problema desses meios é que esses não se atualizam e não fornecem uma resposta rápida e precisa sobre os locais de visitação. Os autores propõem a criação de uma ferramenta para otimizar esse processo, o *Mobile Tourist Guide* (MGT).

Vansteenwegen et al. (2009) utilizaram diferentes heurísticas de busca local para solucionar o problema do *Team Orienteering Problem* (TOP). O TOP é um OP que maximiza a pontuação total das rotas obtidas, sendo que cada rota é limitada a um tempo *Tmax*.

Souffriau et al. (2010), nesse novo trabalho, aplicaram o método *Path Relinking* combinado com um *Greedy Randomized Adaptive Search Procedure* (GRASP) para resolver o problema do *Team Orienteering Problem* (TOP) citado acima. Essa combinação melhorou significativamente os resultados do algoritmo básico do GRASP apresentando melhoras nos resultados obtidos.

Abbaspour e Samadzadegan (2011) propuseram uma solução para o *time-dependent tour planning*. Esse problema consiste em obter uma sequência cronológica de pontos turísticos em um período de tempo por meio de vários meios de transportes diferentes. Utilizando dados reais da cidade de Tehran, localizada no Irã, dois métodos são responsáveis pela formulação das rotas dos turistas: o *Multimodal Shortest Path* fica responsável pela escolha dos melhores caminhos nas ruas da cidade e por determinar quais meios de transporte devem ser usados no percurso; e o *Tour Planning*, que se baseia nos dados informados pelo turista e nas informações obtidas do *Multimodal Shortest Path*, é usado para sugerir uma rota para o turista.

Rodríguez et al. (2012) desenvolveram um sistema baseado na heurística Busca Tabu para rotas de turistas. A ferramenta mostra ao turista qual o é melhor itinerário que satisfaz as suas necessidades de acordo com as informações que o próprio turista insere na aplicação. Sendo assim, a ferramenta retorna ao turista uma rota personalizada atendendo às restrições imposta pelo mesmo.

Ribeiro (2015) propôs um modelo matemático e uma meta-heurística *Simulated Annealing* (SA) para resolução do problema *Tourist Trip Design Problem* (TTDP). O TTDP é conhecido, em português, como Problema de Elaboração de Rotas Turísticas (PERT) e tem como foco a criação de roteiros turísticos respeitando restrições de horários para a visitação, saída e chegada ao hotel. O objetivo considerado pelo autor foi a maximização do somatório de satisfação do turista em sua rota. A satisfação foi calculada de acordo com uma nota dada a cada local que se deseja visitar, sendo que quanto maior a nota maior a satisfação desse turista. Para esse problema, foram utilizados pontos turísticos das cidades de Vila Velha, Guarapari, Cariacica, Viana e Vitória, todas elas no estado do Espírito Santo.

3. METODOLOGIA

A metodologia utilizada neste trabalho consiste na coleta de dados para geração de instâncias para o Problema de Programação de Roteiros Turísticos (PPRT); e para sua resolução, foi utilizada a meta-heurística híbrida *Clustering Search* (CS).

3.1. COLETA DOS DADOS

A coleta dos dados para geração das instâncias referentes ao PPRT foi dividida em duas etapas.

Na primeira etapa, coletou-se via *Google Search* os nomes dos locais que compõem as instâncias, todos os locais e seus dados estão no APÊNDICE A. Foram coletados 115 pontos turísticos do estado do Espírito Santo, considerando todo o Estado em tal pesquisa. Após a definição dos locais, obteve-se, para cada um, suas respectivas coordenadas via *Google Maps*.

A partir dos locais e suas coordenadas, foi possível iniciar a segunda etapa da coleta dos dados. Nessa etapa, desenvolveu-se um código em *Java Script* para fazer a conexão com o *Google Maps* e coletar as informações importantes ao PPRT. Nesse código, as coordenadas de cada local faziam parte de um ponto, e todos os pontos criados formavam um *Array* de locais. O código desenvolvido percorre esse *Array* de locais solicitando ao servidor do *Google Maps* o tempo de deslocamento e a distância entre todos os locais, imprimindo em seguida os resultados na tela do computador.

3.2. GERAÇÃO DE INSTÂNCIAS

A partir dos dados coletados, foram criadas 7 instâncias (problemas testes) distintas a partir dos dados obtidos. A diferenciação dessas instâncias foi baseada na quantidade de locais possíveis para visitação. Assim, foram considerados 25, 50 e 115 locais, a serem visitados por um turista com 7 e 15 dias disponíveis para visitação, gerando então 6 instâncias. Por fim, uma outra instância foi gerada considerando os 115 locais, porém com apenas 3 dias para visitação, simulando assim um possível fim de semana de visitação de um turista com todos os locais disponíveis. No total então, foram geradas 7 instâncias.

Em cada dia disponível para visitação o turista inicia e termina sua rota a partir de um hotel, que também é uma informação considerada nas instâncias. O turista escolhe qual hotel deseja ficar em cada dia, sendo que ele pode sair de um hotel e chegar a outro sem problema algum. Como os hotéis são escolha do turista durante a rota, consideramos que este não vai escolher um hotel para passar a noite fora da cidade a qual estava no ultimo local visitado do dia, ou quando chegou na cidade para visitação do primeiro dia. Para os testes realizados neste trabalho, foi decidido que um hotel estaria de 3 a 60 minutos de um local de visitação, sendo 3 minutos o mais perto, e 60 minutos o mais longe. Essas informações foram geradas aleatoriamente, respeitando o intervalo de 3 a 60 minutos, completando os dados das instâncias.

A Figura 4 ilustra um exemplo de uma instância com 5 locais e 2 dias para visitação. Na primeira linha, o primeiro parâmetro é referente à quantidade de dias que o turista tem para a visitação, o segundo refere-se à quantidade de locais disponíveis e o último parâmetro indica o dinheiro disponível que o turista possui. A segunda e terceira linhas indicam os horários (em minutos) de saída e chegada dos hotéis para cada dia. Neste trabalho, os valores para horário de saída e chegada em cada hotel foram 480 (8:00 da manhã) e 1080 (18:00 da noite), respectivamente. Logo, um dia de visitação inteiro fica restrito ao intervalo de 8:00 da manhã até as 18:00 da noite.

A quarta linha indica os valores do bônus de cada local visitado, ou seja, o bônus refere-se à satisfação do turista quanto a visitar determinado local. Neste trabalho, o bônus de cada local foi definido aleatoriamente, com valores entre 1 e 20. Na quinta e sexta linhas são descritos os horários (em minutos) de abertura e fechamento, respectivamente, de cada local a ser visitado. Esses horários foram definidos aleatoriamente, com valores entre 480 e 1080. A sétima linha contém o tempo mínimo que o turista necessita para visitar determinado local. Neste trabalho, foram usados valores aleatórios entre 30 e 120 minutos. A oitava linha apresenta o valor do ingresso para visitar cada local, também gerados aleatoriamente, com valores entre 10 e 50 reais.

Ao final da instância, tem-se duas matrizes, uma com o tempo de deslocamento entre os locais (em minutos), e a outra com o custo de deslocamento entre os locais (em reais). Os valores da matriz de tempo de deslocamento são os valores

adquiridos na coleta dos dados com os valores de deslocamento para os hotéis. A matriz de custo foi obtida utilizando os valores de distância entre os locais também obtidos na coleta dos dados. Nessa matriz foi considerado que a cada 12 Km gastaria 1 litro de combustível, sendo assim, considerando o valor médio de R$ 3.50 por litro, o turista gasta R$ 0.29 por Km andado. Dessa maneira chegou-se aos dados da matriz de custo.

As matrizes são quadradas de dimensões *L* + *D* + *D*, sendo as *L* primeiras linhas e colunas referentes aos locais de visitação, as próximas *D* referentes aos hotéis de saída em cada dia, e as outras *D* os hotéis de chegada em cada dia. O valor de *L* e *D* indicam a quantidade de locais e dias descritos na primeira linha das instâncias. Os dados do hotel de saída do dia 1 estão na posição *L* + 1, e o hotel de chegada do dia 1 na posição *L* + *D* + 1. A mesma lógica é aplicada para o restante da instância.

```
2       5       100

480     480
1080    1080

7       10      4       1       17
540     480     660     540     480
1020    1080    1080    1020    1080
94      43      115     43      36
37      38      29      32      39

0       17      14      13      14      6       20      9       6
20      0       6       7       8       10      4       3       33
34      9       0       90      201     3       6       16      30
16      6       2       0       2       9       16      10      4
17      6       1       1       0       30      43      27      4
5       60      30      45      59      0       33      59      12
10      12      27      48      3       9       0       43      20
33      20      3       9       12      43      3       0       10
30      6       12      60      33      27      4       27      0

0.00    3.29    2.71    2.51    2.71    1.16    3.87    1.74    1.16
3.87    0.00    1.16    1.35    1.55    1.93    0.77    0.58    8.31
3.29    0.97    0.00    0.19    0.39    0.58    1.16    3.09    5.8
3.09    1.16    0.39    0.00    0.39    1.74    3.09    1.93    0.77
3.29    1.16    0.19    0.19    0.00    5.8     8.31    5.22    0.77
0.96    11.6    5.8     8.7     11.41   0.00    6.38    11.41   2.32
1.93    2.32    5.22    9.28    0.58    1.74    0.00    8.31    3.87
8.31    3.87    0.58    1.74    2.32    11.41   0.58    0.00    1.93
5.8     1.16    2.32    11.6    6.38    5.22    0.77    5.22    0.00
```

Figura 4 - Exemplo de Instância com 5 locais e 2 dias para visitação.

Inicialmente, considerou-se que o turista possui uma quantidade de dinheiro “ilimitada”, com o intuito de verificar o máximo de locais a serem visitados. A partir das soluções obtidas para essas instâncias, definiu-se um valor de dinheiro

disponível para a visitação do turista. Esse valor foi obtido como sendo 90% do valor gasto na melhor solução encontrada para cada instância após 10 execuções da CS descrita na próxima seção. Os valores obtidos foram truncados, ou seja, foram considerados valores inteiros.

3.3. CLUSTERING SEARCH

A *Clustering Search* (CS) foi proposta por Oliveira e Lorena (2004). Os autores chegaram ao CS após fazerem modificações no algoritmo *Evolutionary Clustering Search* (ECS), tendo como intenção a generalização do mesmo. A ideia foi substituir o algoritmo evolutivo por outras meta-heurísticas. A CS consiste basicamente em obter um conjunto de soluções e aplicar sobre elas uma busca intensificada nas regiões consideradas mais promissoras.

A escolha da CS se deu pelo fato da mesma apresentar bons resultados em trabalhos de otimização combinatória. Sua estrutura fornece uma boa opção para testes, pois a mesma permite que a heurística geradora de soluções possa ser trocada, fazendo com que se possa testar outras combinações e verificar qual melhor se adequa ao PPRT. A Figura 5 ilustra o fluxograma de execução da CS.

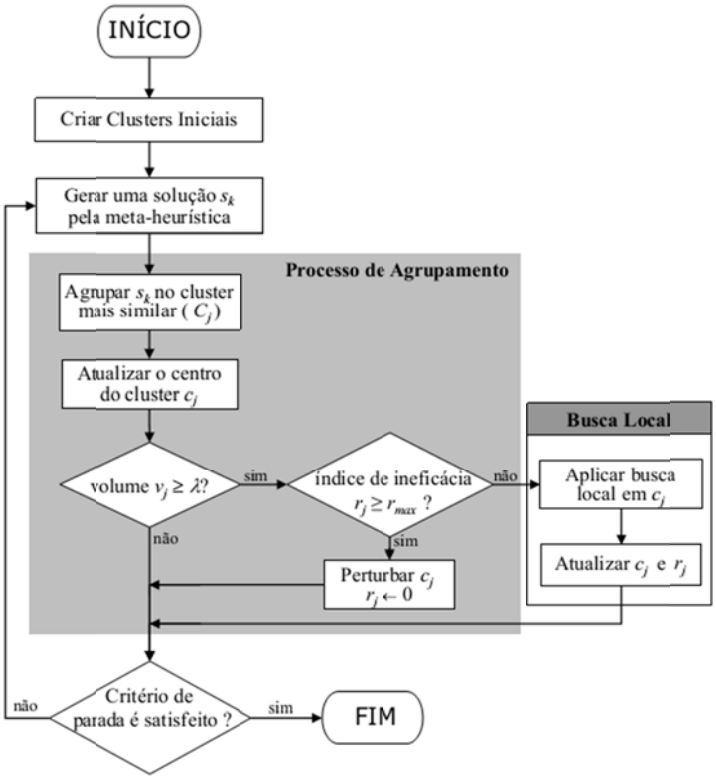

Figura 5 - Fluxograma da CS

Fonte: Chaves e Lorena (2010).

Segundo Chaves e Lorena (2010), a CS é um método iterativo que agrupa várias soluções em *clusters*, dividindo o espaço de busca e utilizando localização de regiões consideradas promissoras. Um *cluster* pode ser estabelecido como um conjunto de três elementos $C = \{c, v, r\}$. Sendo assim, um *cluster i* é representado pelo seu centro c_i, que marca sua posição dentro da região de busca. A variável v_i indica o volume do *cluster i*, ou seja, a quantidade de soluções associadas ao *cluster i*. Caso v_i tenha alcançado o valor máximo de soluções λ, isso significa que que essa região se tornou promissora, e então seu índice de ineficácia r_i deverá ser analisado. O índice de ineficácia r_i é utilizado para dizer se o centro do cluster *i* está ou não sendo melhorado pela função de busca local. Se a busca local foi aplicada sobre o *cluster i* e este não foi melhorado, o valor de r_i será incrementado para indicar o número de vezes em que a busca local não foi eficaz. O parâmetro r_{max} indica o valor máximo que r_i pode alcançar. Quando r_i alcança o valor de r_{max}, uma perturbação é aplicada sobre o centro do *cluster*. Dessa maneira, é evitado que regiões que já tenham sido muito exploradas pelo algoritmo de busca local sejam novamente consideradas. Os principais componentes da CS são descritos a seguir.

3.3.1. Principais Componentes da CS

3.3.1.1. Gerador de soluções

Para este trabalho, foi utilizada a *Simulated Annealing* (SA) como meta-heurística geradora de solução.

A *Simulated Annealing* (SA) foi apresentada por Kirkpatrick et al. (1983) como uma técnica de otimização combinatória utilizada em sistemas eletrônicos. A ideia por trás da SA faz uma analogia ao processo de resfriamento de metal em seu estado de fusão. Sendo uma meta-heurística muito eficaz, ela continua sendo muito utilizada, tanto individualmente como em conjunto com outras meta-heurísticas. Mais detalhes sobre o SA podem ser vistos em Ribeiro et al. (2011). A Figura 6 apresenta o pseudocódigo da SA.

```
1.  DADO(α, SA_max, T_0, T_C e S) FAÇA
2.  s* ← s   T ← T_0
3.  ENQUANTO T > T_C FAÇA
4.     iter ← 0
5.     ENQUANTO iter < SA_max FAÇA
6.        iter ← iter + 1
7.        s' ← N(s)
8.        SE f(s') > f(s) ENTÃO
9.           s ← s'
10.       SENÃO
11.          Com probabilidade e^(-(f(s)-f(s'))/T) s ← s'
12.       FIM-SE
13.    FIM-ENQUANTO
14.    T ← αT
15. FIM-ENQUANTO
16. Retornar s*
```

Figura 6 - Pseudocódigo da SA.

A SA recebe 5 parâmetros de entrada, sendo α a taxa de resfriamento, SA_{max} o número máximo de iterações, T_0 a temperatura inicial, T_c a temperatura de congelamento e S uma solução inicial qualquer. A solução s^* armazena a melhor solução encontrada durante a execução do algoritmo. A vizinhança $N(s)$ (linha 7 da Figura 6) utilizada neste trabalho encontra uma solução s' próxima da solução atual s.

Um dos parâmetros de entrada do SA é a solução inicial. Uma solução para o PPRT é representada na Figura 7. A solução é formada por um vetor de dias, e cada dia possui um vetor de locais, que são números inteiros de 1 a 115 (no caso de uma instância com 115 locais), representando todos os 115 locais obtidos na coleta dos dados.

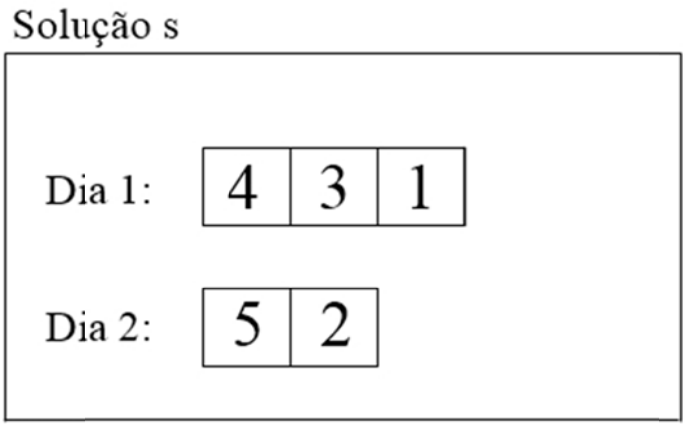

Figura 7 - Representação de uma solução para o PPRT.

Na Figura 7, *s* é uma solução que possui 2 dias de visitação, e cada dia possui sua sequência de locais a ser visitado. No primeiro dia, o turista visitará o local 4, depois se deslocará do 4 até o 3, e assim por diante. A Figura 8 ilustra um exemplo de rota para o primeiro dia da solução apresentada na Figura 7. Na Figura 8, o Convento da Penha representa o local 4, a Fábrica de Chocolates Garoto representa o local 3 e, por fim, a praia da costa representa o local 1. Todos esses locais são da cidade de Vila Velha no Espírito Santo.

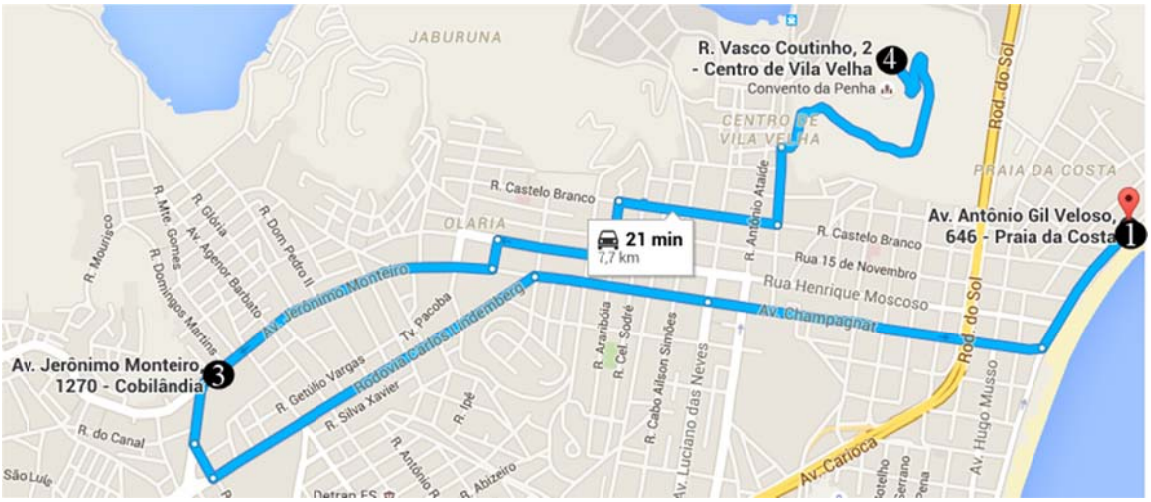

Figura 8 - Exemplo de uma Rota no *Google Maps*

A solução inicial para a SA é construída colocando todos os *L* locais aleatoriamente dentro da solução. Primeiro, é sorteado um local e um dia aleatoriamente. Feito isso, adiciona-se o local no dia sorteado. Isso é repetido até que todos os 115 locais estejam dentro da solução. Essa solução inicial é então passada como entrada para a SA. A SA utiliza movimentos para encontrar vizinhos ($N(s)$ - linha 7 da Figura 6), que são soluções próximas da solução corrente. Os vizinhos podem ser soluções melhores ou piores do que a solução atual, ou seja, os vizinhos são utilizados para procurar novas soluções. Para alcançar a maioria dos vizinhos possíveis, são aplicados movimentos que modificam uma parte de uma solução atual formando uma nova solução, a qual será então comparada com a solução atual.

Para o PPRT, foram criados 5 movimentos possíveis. Utilizando a Figura 7 como base, o primeiro movimento consiste em trocar dois locais de lugar em um mesmo dia, ou seja, seria como trocar de lugar o local 4 com o 1, no primeiro dia por exemplo. O segundo movimento troca dois locais em dias diferentes. Considerando

a Figura 7, seria como trocar o local 4 do dia 1 com o local 2 do dia 2. O terceiro movimento remove um local de um dia. Como exemplo, seria retirar o local 5 do dia 2. O quarto movimento insere um local que não está dentro da solução em um dia qualquer. Um exemplo seria como colocar o local 100 no dia 2. Toda solução *s* utiliza um vetor auxiliar para armazenar os locais que não estão sendo visitados na solução. Por fim, o movimento 5 retira um local de um dia e coloca em outro dia. Como exemplo seria como se o local 3 do dia 1 fosse colocado no dia 2.

3.3.1.2. Agrupamento iterativo

O agrupamento iterativo tem como objetivo alocar soluções ao *cluster* mais similar. O centro c_i do *cluster* é atualizado, ou seja, a solução antiga é substituída por uma nova fazendo com que ele sofra um distúrbio. A partir de um número γ de *clusters*, o SA melhora sua solução inicial *s* e atribuí ao cluster *i* que possua maior semelhança a solução corrente. Tal semelhança se dá pela distância de Hamming H_i (HAMMING, 1950), que neste trabalho é calculada de acordo com o número de desigualdades encontradas entre a solução *s* e o centro do *cluster*. Um exemplo do cálculo da distância de Hamming é apresentado na Figura 9.

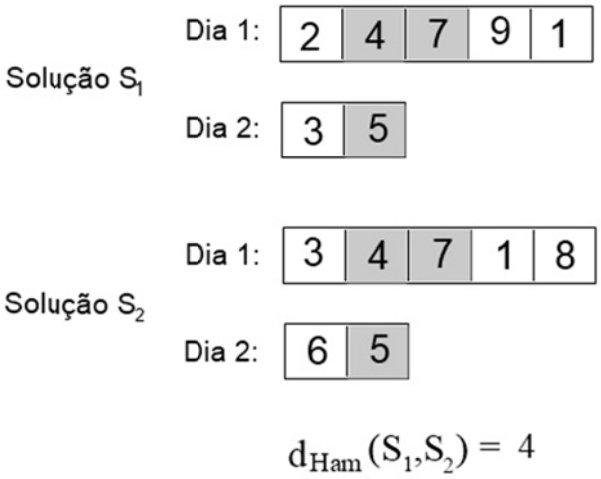

Figura 9 - Exemplo de cálculo da distância de Hamming.

Na Figura 9, as diferenças entre as soluções S_1 e S_2 são as "células" brancas, ou seja, existem 4 valores distintos. Logo, esse valor é a distância de Hamming entre S_1 e S_2. Ao final do processo de agrupamento, a solução *s* é atribuída ao *cluster i* que esteja menos distante, ou seja, com o menor valor de d_{Ham}. Para isso, é utilizada a

“assimilação direta”, ou seja, se uma solução *s* for melhor do que o centro do *cluster* atual (c_i), esta passará a ser o novo centro do *cluster*.

3.3.1.3. Analisador de agrupamentos

A função do analisador de agrupamentos é fazer um exame em cada *cluster*, observando se seu volume v_i alcançou o limite λ, que já havia sido estabelecido previamente. Quando λ é atingido, significa que houve muitas soluções semelhantes naquele *cluster* e, sendo assim, torna-se promissor, indicando que precisa ser melhor analisado.

Caso o *cluster* não esteja tendo uma melhora, ou seja, a busca local já tenha sido usada mais de r_{max} vezes em determinado *cluster i*, uma perturbação é realizada, permitindo assim escapar de ótimos locais. Por outro lado, se $r_i < r_{max}$, o algoritmo de busca local será aplicado no centro c_i do *cluster*, explorando assim uma suposta região considerada promissora. Essa também é uma função do analisador de agrupamentos. A perturbação foi realizada por meio da aplicação de um dos movimentos de geração de vizinhos do SA, descritos na Seção 3.3.1.1. A escolha do movimento a ser utilizado é feita de forma aleatória.

3.3.1.4. Busca local

Foi implementada uma busca loca de melhor melhora (HANSEN; MLADENOVIC, 2006). Nessa busca, cada local que está dentro da solução é trocado com todos os locais que estão fora da solução e, após todas as trocas, aquela que resultar na melhor solução é aceita, e toda a busca é recomeçada a partir dessa melhor solução. No caso da Figura 7, por exemplo, o local 4 é trocado com todos os locais que não estão na solução (10, 19, 100, etc.), depois o local 3, e assim por diante até o local 2 (último local no segundo dia).

3.3.1.5. Função objetivo

A função objetivo (FO) visa maximizar a satisfação do turista após a criação de todo roteiro turístico. No presente trabalho, foi utilizada a Equação (1) para descrever a função objetivo para uma solução *s* do PPRT. São utilizados pesos para penalizar as violações de cada uma das restrições do problema. Para cada violação é multiplicado um valor w_i correspondente à violação ocorrida na restrição *i*. Assim, as

violações penalizadas são subtraídas da “satisfação” do turista. Caso alguma restrição seja violada exceto a *VLA*, a solução será inviável para o problema, e será subtraído um valor na função objetivo. Caso contrário, ou seja, nenhuma restrição é violada, o valor da função objetivo será exatamente a satisfação do turista.

$$f(s) = Bônus - (w_1 * VLD - w_2 * VLA - w_3 * VH - w_4 * VD) \quad (1)$$

Bônus é o somatório de satisfação do turista para cada local visitado na solução *s*. w_1 é o peso da *VLD*, que é a violação por chegar depois que o local fechou. w_2 é o peso da *VLA*, que é a violação por chegar antes do local estar aberto. w_3 é o peso de *VH* que é a violação de chegada depois do horário limite para o dia do turista, ou seja, horário que o turista deveria retornar ao seu hotel no final do dia. Por fim, w_4 é o peso de violação da restrição de dinheiro *VD*, ou seja, se for gasto mais dinheiro do que o limite definido pelo turista.

Para o PPRT em questão, chegar antes do local estar aberto não é tão prejudicial, pois o turista pode esperar o local abrir, ou seja, isso não inviabiliza uma solução, porém deve ser evitado. Já as demais restrições inviabilizam a solução. Por isso, os pesos foram definidos empiricamente, sendo: $w_1 = 500$, $w_2 = 1$, $w_3 = 500$, e $w_4 = 500$.

3.3.2. Algoritmo CS

O primeiro passo da CS é criar γ *clusters* aleatoriamente, considerando soluções iniciais geradas conforme descrito na Seção 3.3.1.1. Inicia-se o volume dos *clusters* v_i e o índice de ineficácia r_i. A solução *s* e a melhor solução *s** recebem uma solução inicial passada pelo SA. A SA é executada e após o resfriamento, a solução atual é enviada à CS para o processo de agrupamento. A solução atual *s* gerada pela SA é associada ao *cluster* mais similar, e seu volume e centro são atualizados. É verificado então o índice de ineficácia, e a necessidade de aplicação da busca local. Por fim, verifica-se se a nova solução c_i (centro do *cluster*) é melhor do que a melhor solução geral *s**. A execução da SA é continuada, e todo o algoritmo CS repetido até que um critério de parada seja alcançado. O pseudocódigo da CS com a SA utilizada neste trabalho é apresentado na Figura 10.

```
Criar γ novas soluções (clusters) aleatoriamente
v_i ← 1 e r_i ← 0 ∀ i = 1,...,γ
s ← solução inicial
s* ← s  T ← T_0
ENQUANTO T > T_C FAÇA
iter ← 0
ENQUANTO iter < SA_max FAÇA
   iter ← iter + 1
   s' ← N(s)
   SE f(s') > f(s) ENTÃO
      s ← s'
   SENÃO
      Com probabilidade e^(-(f(s)-f(s'))/T) s ← s'
   FIM-SE
FIM-ENQUANTO
T ← αT
i ← arg min_{i∈{1,...,γ}} {H_i}
v_i ← v_i + 1
c_i ← melhor(s,c_i)
SE v_i = λ ENTÃO
   v_i ← 0
   s ← busca local(c_i)
   SE f(s) = f(c_i) ENTÃO
      r_i ← r_i + 1
      SE r_i = r_max ENTÃO
         r_i ← 0
         c_i ← N(c_i)
      FIM-SE
   SENÃO
      r_i ← 0
      c_i ← s
   FIM-SE
FIM-SE
s* ← max(s*,c_i)
FIM-ENQUANTO
Retornar s*
```

Figura 10 - Pseudocódigo da CS implementada.

4. RESULTADOS

Para calibração dos parâmetros da *Simulated Annealing* (SA) e da *Clustering Search* (CS) foram executados testes com diferentes combinações de parâmetros. Primeiro, foi feito a calibração da SA, sendo que, para cada combinação de parâmetros foram executados 3 testes utilizando a instância com 115 locais e 7 dias de visitação. Essa instância foi escolhida aleatoriamente. A partir das três soluções obtidas para essa instância, foram extraídos os dados apresentados na Tabela 1.

Tabela 1 - Calibração da SA.

SA_{max}	α	T_0	T_C	FO Média	Melhor FO	Desvio (%)	Tempo Médio (seg.)
1000	0,975	20000	0,01	668,00	686	2,62	71,33
1000	0,975	10000	0,01	682,67	698	2,20	66,33
1000	0,975	1000	0,01	678,33	687	1,26	54,67
10000	0,975	20000	0,01	684,67	694	1,34	618,00
10000	0,975	10000	0,01	700,00	703	0,43	523,67
10000	0,975	1000	0,01	706,00	714	1,12	443,67
1000	0,995	20000	0,01	697,67	717	2,70	313,33
1000	**0,995**	**10000**	**0,01**	**708,00**	711	**0,42**	308,00
1000	0,995	1000	0,01	705,33	715	1,35	243,67
10000	0,995	20000	0,01	397,67	495	19,66	2132,00
10000	0,995	10000	0,01	414,67	466	11,02	1516,67
10000	0,995	1000	0,01	361,67	372	2,78	1497,67

Na Tabela 1, $SA_{\max}$ indica a quantidade de iterações da SA (1000 ou 10000), α é a taxa de resfriamento (0,975 ou 0,995), T_0 é a temperatura inicial (20000, 10000 ou 1000) e T_c é a temperatura de congelamento (0,01), ou seja, esses são os parâmetros da SA. Os valores utilizados foram definidos empiricamente. Para cada uma dessas combinações, foram executados 3 testes com 10 minutos de duração cada. Ao final de todas as execuções, foi possível chegar a uma conclusão observando os resultados obtidos: a coluna *FO Média* indica a média aritmética entre as três FOs obtidas, sendo a FO calculada de acordo com a Equação (1), ou seja, FO = $f(s)$. O valor da melhor FO entre os três testes realizados é apresentada na coluna *Melhor FO*, e o Desvio entre as três soluções é calculado de acordo com a seguinte fórmula: *Desvio* = (((*Melhor FO* – *FO Média*) / *Melhor FO*) * 100). Por fim, a coluna *Tempo Médio* indica o tempo que a SA levou para encontrar a melhor solução.

A partir dos resultados apresentados na Tabela 1, definiu-se os valores dos parâmetros para a SA considerando aqueles que obtiveram a melhor média para a FO e o melhor desvio (destacados em negrito).

Para a calibração da CS, foi utilizada a mesma ideia da SA, ou seja, foram executados 3 testes com 10 minutos utilizando a mesma instância para cada combinação de parâmetros. Os parâmetros utilizados pela SA foram aqueles obtidos após a sua calibração (Tabela 1). A Tabela 2 apresenta os resultados da calibração de parâmetros para a CS.

Tabela 2 - Calibração da CS.

γ	λ	r_{max}	FO Média	Melhor FO	Desvio (%)	Tempo Médio (seg.)
10	5	3	687,67	701	1,90	317,65
10	5	5	688,67	710	3,00	360,15
10	5	7	696,33	707	1,51	320,91
10	7	3	688,00	704	2,27	322,79
10	7	5	682,00	690	1,16	320,08
10	7	7	689,67	722	4,48	314,90
10	9	3	694,00	708	1,98	303,34
10	9	5	702,33	708	0,80	296,91
10	9	7	677,67	694	2,35	292,28
7	5	3	693,33	718	3,44	289,29
7	5	5	692,67	698	0,76	348,81
7	5	7	697,00	706	1,27	333,43
7	7	3	669,33	688	2,71	287,89
7	7	5	692,33	694	0,24	303,03
7	7	7	693,33	697	0,53	294,82
7	9	3	703,33	721	2,45	281,85
7	9	5	695,67	704	1,18	301,76
7	9	7	693,67	702	1,19	296,21
5	5	3	693,67	699	0,76	294,35
5	5	5	690,00	709	2,68	302,35
5	**5**	**7**	**703,67**	**724**	2,81	343,32
5	7	3	682,00	691	1,30	303,88
5	7	5	686,00	698	1,72	278,86
5	7	7	699,00	706	0,99	301,10
5	**9**	**3**	**705,67**	**711**	0,75	293,78
5	9	5	674,33	704	4,21	298,58
5	9	7	690,67	699	1,19	292,24

Na Tabela 2, γ indica a quantidade máxima de *clusters* (10, 7 ou 5), λ o volume máximo de soluções em cada *cluster* (5, 7 ou 9) e r_{max} o limite para o índice de ineficácia (3, 5 ou 7), formando assim todos os parâmetros da CS. A partir dos resultados apresentados na Tabela 2, decidiu-se por duas combinações (em destaque), pois ficaram muito parecidas quando julgado seus resultados. Logo, foi feito um desempate com o intuito de definir o melhor conjunto de valores para os parâmetros. A CS foi executada 10 vezes por 10 minutos para cada combinação de parâmetros. Os resultados obtidos são apresentados na Tabela 3.

Tabela 3 - Desempate da Calibração da CS.

γ	λ	r_{max}	**Média da FO**	**Melhor FO**	**Desvio**	**Tempo Médio**
5	**5**	**7**	**684,30**	708	**3,35**	309,40
5	9	3	683,40	711	3,88	296,08

Utilizando a mesma ideia das calibrações anteriores, é possível observar que os dados não ficaram tão distantes um dos outros. Porém, observando o *Desvio* e a *FO Média*, optou-se pelo primeiro conjunto de parâmetros para a CS. Essa escolha foi baseada no fato de que um desvio menor e uma média maior levam o método a ter uma robustez maior. A partir da calibração da SA e da CS, os parâmetros ficaram então definidos de acordo com a Tabela 4.

Tabela 4 - Melhores parâmetros para a SA e a CS.

SA	SA_{max}	1000
	α	0,995
	T_0	10000
	T_C	0,01
CS	γ	5
	λ	5
	r_{max}	7

Utilizando os valores obtidos para os parâmetros da SA e da CS (Tabela 4), a CS foi executada 10 vezes por 10 minutos para cada uma das 7 instâncias geradas (descritas na Seção 3.2). A Tabela 5 apresenta, em detalhes, a melhor solução obtida para cada instância. A coluna *Instância* indica a instância utilizada, com a quantidade de locais e dias, *FO* é a Função objetivo de cada solução, *Dinheiro Gasto* é o dinheiro que foi gasto durante todo o roteiro turístico, *Bônus Total* é a

satisfação total do turista ao visitar todos os locais da rota criada na solução, *Tempo Melhor* é o tempo para achar a melhor solução, *Tempo Total* é o tempo total de execução da CS, *Qtde de Locais Visitados* é a quantidade de locais que foram visitados pelo turista, *Qtde de Locais NÃO Visitados* é a quantidade de locais que não foram visitados pelo turista e *Dias Utilizados* indica a quantidade de dias que foram utilizados para fazer as rotas.

Tabela 5 - Melhores soluções SEM a restrição de dinheiro.

Instância	FO	Dinheiro Gasto	Bônus Total	Tempo Melhor (seg.)	Tempo Total (seg.)	Qtde de Locais Visitados	Qtde de Locais NÃO Visitados	Dias Utilizados
25 locais e 7 dias	297	1130,99	297	56,36	600,01	25	0	7
25 locais e 15 dias	297	1148,78	297	0,05	600,01	25	0	12
50 locais e 7 dias	510	1403,56	510	219,60	600,01	39	11	7
50 locais e 15 dias	551	2161,83	551	174,02	600,01	50	0	14
115 locais e 7 dias	719	1615,44	719	232,86	600,00	46	69	7
115 locais e 15 dias	1146	3210,85	1146	264,26	600,00	92	23	15
115 locais e 3 dias	377	785,54	377	217,05	600,00	23	92	3

A partir das soluções apresentada na Tabela 5, definiu-se um limite de dinheiro para o turista, conforme descrito na Seção 3.2. A partir de então, a CS foi executada novamente por 10 vezes por 10 minutos para cada uma das 7 instâncias geradas, porém considerando agora um limite de dinheiro. As melhores soluções obtidas são apresentadas na Tabela 6. Nessa tabela, a coluna *Dinheiro disponível* indica o dinheiro total que o turista tinha para gastar.

Tabela 6 - Melhores soluções COM a restrição de dinheiro.

Instância	FO	Dinheiro Disponível	Dinheiro Gasto	Bônus Total	Tempo Melhor (seg.)	Tempo Total (seg.)	Qtde de Locais Visitados	Qtde de Locais NÃO Visitados	Dias Utilizados
25 locais e 7 dias	297	904	902,25	297	172,97	600,01	25	0	7
25 locais e 15 dias	297	942	939,17	297	0,98	600,01	25	0	14
50 locais e 7 dias	482	1263	1262,06	482	195,38	600,01	35	15	7
50 locais e 15 dias	551	1921	1920,37	551	190,44	600,01	50	0	15
115 locais e 7 dias	688	1453	1447,85	688	223,49	600,01	43	72	7
115 locais e 15 dias	1120	2889	2886,55	1120	243,01	600	84	31	15
115 locais e 3 dias	352	706	670,82	352	176,22	600,01	20	95	3

Como pode ser observado nas Tabelas 5 e 6, os resultados obtidos mostram um comportamento diferente quando a restrição de dinheiro é considerada. Sem a restrição de dinheiro (Tabela 5), a CS tenta colocar o máximo de locais dentro da solução, até ser impedido pela restrição de tempo, ou seja, enquanto tem tempo no dia de visitação a CS encaixa os locais respeitando suas restrições. Já quando se considera a restrição de dinheiro (Tabela 6), a sequência de visitação fica mais "importante", pois ela influencia diretamente no "custo" para o turista. Assim, a CS tenta encaixar os locais de forma a criar uma boa ordem de visitação, buscando reduzir o dinheiro gasto. Observando as duas primeiras soluções das Tabelas 5 e 6, percebe-se que há uma grande economia no dinheiro gasto na visitação quando se considera a restrição de dinheiro (Tabela 6), sendo que todos os locais foram visitados em todas as soluções. Porém, a sequência de visitação obtida pelas soluções considerando a restrição de dinheiro proporcionou um gasto reduzido, economizando cerca de 20% nessas duas soluções. No geral, considerando todas as sete instâncias, houve uma economia de 12,46%.

As Tabelas 7 e 8 apresentam uma visão geral de todas as execuções da CS sem e com a restrição de dinheiro, respectivamente. A coluna *Menor Dinheiro* indica o menor valor gasto pelo turista entre as 10 soluções obtidas para cada instância. As demais colunas são similares àquelas apresentadas nas tabelas anteriores.

Tabela 7 - Visão geral das execuções SEM a restrição de dinheiro.

Instância	**FO Média**	**Melhor FO**	**Desvio (%)**	**Tempo Médio (seg.)**	**Menor Dinheiro**
25 locais e 7 dias	297,00	297	0,00	134,06	1004,74
25 locais e 15 dias	297,00	297	0,00	0,10	1047,29
50 locais e 7 dias	502,10	510	1,55	199,12	1343,65
50 locais e 15 dias	551,00	551	0,00	176,19	2134,78
115 locais e 7 dias	692,50	719	3,69	212,10	1428,96
115 locais e 15 dias	1134,10	1146	1,04	244,79	3035,15
115 locais e 3 dias	347,80	377	7,75	193,98	682,62

Tabela 8 - Visão geral das execuções COM a restrição de dinheiro.

Instância	FO Média	Melhor FO	Desvio (%)	Tempo Médio (seg.)	Dinheiro disponível	Menor Dinheiro
25 locais e 7 dias	297	297	0,00	183,17	904	901,89
25 locais e 15 dias	297	297	0,00	24,32	942	933,95
50 locais e 7 dias	478,80	482	0,66	203,69	1263	1251,94
50 locais e 15 dias	551	551	0,00	200,20	1921	1919,79
115 locais e 7 dias	672,60	688	2,24	209,67	1453	1423,13
115 locais e 15 dias	1107,70	1120	1,10	235,15	2889	2876,30
115 locais e 3 dias	335,30	352	4,74	193,27	706	619,24

5. CONCLUSÕES

Este trabalho apresentou uma alternativa para resolução do Problema de Programação de Roteiros Turísticos (PPRT), que tem como objetivo criar roteiros turísticos para diferentes tipos de turista, levando em consideração seu tempo e dinheiro disponível, e respeitando as restrições impostas ao problema.

Para a resolução do PPRT, foi proposta a utilização da meta heurística híbrida *Clustering Search* (CS) com a *Simulated Annealing* (SA) como geradora de soluções. A CS vem apresentando bons resultados para problemas de natureza combinatorial, e se mostrou eficiente na resolução do PPRT.

Os resultados obtidos demonstram o comportamento do método para diferentes situações, sendo aplicado a duas categorias de instâncias: sem restrição de dinheiro e com restrição de dinheiro. O método proposto se comportou diferente para cada categoria, produzindo resultados de acordo com cada categoria e respeitando as restrições que lhe foram impostas.

Dessa forma, fica claro que a CS pode ser considerada uma nova e boa alternativa para resolução do PPRT, podendo ser explorada para novas instâncias com locais de visitação diferentes, considerando, por exemplo, outros estados e até mesmo países diferentes.

REFERÊNCIAS BIBLIOGRÁFICAS

ABBASPOUR, R. A.; SAMADZADEGAN, F. Time-dependent personal tour planning and scheduling in metropolises. *Expert Systems with Applications*, v. 38, n. 10, p. 12439-12452, 2011.

CHAO, I-M.; GOLDEN, B. L.; WASIL, E. A. The team orienteering problem. *European Journal of Operational Research*, v. 88, n. 3, p. 464-474, 1996.

CHAVES, A. A.; LORENA, L. A. N. Clustering search algorithm for the capacitated centered clustering problem. *Computers & Operations Research*, v. 37, p. 552-558, 2010.

GOLDEN, Bruce L.; LEVY, Larry; VOHRA, Rakesh. The orienteering problem. Naval research logistics, v. 34, n. 3, p. 307-318, 1987.

HAMMING, R. W. Error detecting and error correcting codes. *Bell System Technical Journal*, v. 26, p. 147-160, 1950.

HANSEN, P.; MLADENOVIC, N. First vs. best improvement: an empirical study. *Discrete Applied Mathematics*, v. 154, p. 802-817, 2006.

KIRKPATRICK, S.; GELATT, C.; VECCHI, M. Optimization by simulated annealing. *Science*, v. 220, n. 4598, p. 498-516, 1983.

OLIVEIRA, A. C. M; LORENA, L. A. N. Detecting promising areas by evolutionary clustering search. *Lecture Notes in Artificial Intelligence*, v. 3171, p. 385-394, 2004.

RIBEIRO, E. A. Modelo matemático e meta-heurística simulated annealing para elaboração de roteiros turísticos com base no *tourist trip design problem with time window*. Dissertação (Mestrado em Gestão Pública) – Universidade Federal do Espírito Santo, Vitória, 2015.

RIBEIRO, G. M.; MAURI, G. R.; LORENA, L. A. N. A simple and robust simulated annealing algorithm for scheduling workover rigs on onshore oil fields. *Computers & Industrial Engineering*, v. 60, n. 4, p. 519-526, 2011.

RODRÍGUEZ, B; MOLINA, J.; PÉREZ, F. CABALLERO, R. Interactive design of personalized tourism routes. *Tourism Management*, v. 33, n. 4, p. 926-940, 2012.

VANSTEENWEGEN, P.; SOUFFRIAU, W.; BERGHE, G. V.; OUDHEUSDEN, D. V. A guided local search metaheuristic for the team orienteering problem. *European Journal of Operational Research*, v. 196, n. 1, p. 118-127, 2009.

SOUFFRIAU, W.; VANSTEENWEGEN, P.; BERGHE, G. V.; OUDHEUSDEN, D. V. A path relinking approach for the team orienteering problem. *Computers & Operations Research*, v. 37, n. 11, p. 1853-1859, 2010.

VANSTEENWEGEN, P.; OUDHEUSDEN, D. V. The mobile tourist guide: an OR opportunity. *OR Insight*, v. 20, n. 3, p. 21-27, 2007.

APÊNDICE A. LISTA COMPLETA DOS PONTOS TURÍSTICOS

A Tabela 9 contém todos os pontos turísticos coletados (descritos na Seção 3.1), sendo que a coluna *Ponto Turístico* é o nome do ponto turístico e a coluna *Cidade* é a cidade em que esse ponto turístico se encontra. Já a coluna *Índice* é o número utilizado para representar cada ponto turístico nas instâncias geradas, e *Ponto no Google Maps* é a coordenada de cada ponto turístico obtida a partir do *Google Maps*.

Tabela 9 - Pontos turísticos

Ponto Turístico	Cidade	Índice	Ponto no *Google Maps*
Associação Paneleiras de Goiabeiras	Vitória	1	-20.269655, -40.303175
Cais do Hidroavião	Vitória	2	-20.311648, -40.358652
Capela de Santa Luzia	Vitória	3	-20.319631, -40.338439
Casa Porto das Artes Plásticas	Vitória	4	-20.319562, -40.336142
Catedral Metropolitana de Vitória	Vitória	5	-20.319907, -40.337180
Convento de São Francisco	Vitória	6	-20.318700, -40.339640
Curva da Jurema	Vitória	7	-20.309147, -40.287852
Escola de Teatro e Dança Fafi	Vitória	8	-20.319812, -40.333337
Estação Porto – Armazém 5	Vitória	9	-20.321904, -40.337638
Hortomercado	Vitória	10	-20.316876, -40.299055
Igreja e Convento do Carmo	Vitória	11	-20.318328, -40.338091
Igreja Nossa Senhora da Glória	Barra do Jucu	12	-20.325171, -40.278627
Igreja Nossa Senhora do Rosário	Barra do Jucu	13	-20.330579, -40.291996
Igreja de São Gonçalo	Vitória	14	-20.320071, -40.339416
Ilhas das Caieiras	Vitória	15	-20.277920, -40.335531
Mosteiro Zen Budista	Ibiraçu	16	-19.890422, -40.377814
Museu de Artes do Espírito Santo	Vitória	17	-20.319939, -40.334417
Museu Ferroviário	Vila Velha	18	-20.326599, -40.345133
Museu Solar Monjardim	Vitória	19	-20.305822, -40.321886
Observatório Astronômico	Vitória	20	-20.276795, -40.303998
Palácio Anchieta	Vitória	21	-20.321142, -40.339808
Parque da Fonte Grande	Vitória	22	-20.298139, -40.332903
Parque Moscoso	Vitória	23	-20.319313, -40.341965
Parque Municipal Gruta da Onça	Vitória	24	-20.317604, -40.329541
Parque Municipal Horto de Maruipe	Vitória	25	-20.296086, -40.313597
Parque Municipal Pedra da Cebola	Vitória	26	-20.277168, -40.299680
Passeio de Escuna Cores do Mar	Vitória	27	-20.292043, -40.286153
Pedra dos Dois Olhos	Vitória	28	-20.301519, -40.328781
Planetário de Vitória	Vitória	29	-20.277592, -40.307093
Praia de Camburi	Vitória	30	-20.280087, -40.286897

Praça Costa Pereira	Vitória	31	-20.320187, -40.335649
Praça da Ciência	Vitória	32	-20.307483, -40.291583
Praça dos Desejos	Vitória	33	-20.306497, -40.291202
Praça dos Namorados	Vitória	34	-20.298848, -40.290969
Santuário de Santo Antônio	Vitória	35	-20.308344, -40.352765
Sítio Histórico da Prainha	Vitória	36	-20.329115, -40.292604
Teatro Carlos Gomes	Vitória	37	-20.319482, -40.335076
Triângulo das Bermudas	Vitória	38	-20.297820, -40.293021
Casa de Pedra	Serra	39	-20.149192, -40.186210
Casa do Congo “Mestre Antônio Rosa”	Serra	40	-20.127029, -40.308837
Clube Capixaba de Golf	Serra	41	-20.200725, -40.272989
Igreja e Residência de Reis Magos	Serra	42	-20.058378, -40.192260
Igreja Nossa Senhora da Conceição	Serra	43	-20.129173, -40.307807
Lagoas	Serra	44	-20.106325, -40.246315
Manguinhos	Serra	45	-20.193333, -40.198434
Mestre Álvaro	Serra	46	-20.240344, -40.274906
Museu Histórico da Serra	Serra	47	-20.127917, -40.308276
Oficina de Instrumentos de Congo	Serra	48	-20.136680, -40.301220
Santuário das Formigas Bordadeiras	Serra	49	-20.132521, -40.306724
Sítio Histórico de Carapina	Serra	50	-20.221879, -40.280173
Convento da Penha	Vila Velha	51	-20.328448, -40.287300
Fábrica de Chocolates Garoto	Vila Velha	52	-20.338092, -40.308958
Farol Santa Luzia	Vila Velha	53	-20.324761, -40.268334
Gruta do Frei Palácios	Vila Velha	54	-20.329989, -40.289170
Morro do Moreno	Vila Velha	55	-20.326233, -40.279086
Praia da Costa	Vila Velha	56	-20.333562, -40.279359
Praia Itaparica	Vila Velha	57	-20.370278, -40.302443
Praia Itapuã	Vila Velha	58	-20.351547, -40.283745
Casa da Cultura	Guarapari	59	-20.671046, -40.521824
Enseada Azul	Guarapari	60	-20.724978, -40.525081
Gruta de Sant'ana	Guarapari	61	-20.668467, -40.496052
Parque Estadual Paulo Cézar Vinha	Guarapari	62	-20.586175, -40.410354
Poço dos Jesuitas	Guarapari	63	-20.666584, -40.494458
Praia da Areia Preta	Guarapari	64	-20.674665, -40.499764
Praia das Castanheiras	Guarapari	65	-20.672232, -40.497129
Praia de Meaípe	Guarapari	66	-20.738024, -40.540262
Praia do Morro	Guarapari	67	-20.653781, -40.486597
Praia dos Namorados	Guarapari	68	-20.671201, -40.495146
Ruínas da Igreja	Guarapari	69	-20.669763, -40.499246
Mochuara	Cariacica	70	-20.326850, -40.393068
Congo de Máscaras	Cariacica	71	-20.313309, -40.439214

Reserva Biológica de Duas Bocas	Cariacica	72	-20.264285, -40.420359
Casa da Cultura	Viana	73	-20.390409, -40.496797
Cachoeira da Fumaça	Alegre	74	-20.631711, -41.604006
Rampa do Urubu	Viana	75	-20.356547, -40.448436
Museu Nacional Padre Anchieta	Anchieta	76	-20.804217, -40.653684
Praia Anchieta	Anchieta	77	-20.807699, -40.646762
Praia Balanço	Anchieta	78	-20.809795, -40.663852
Praia Castelhanos	Anchieta	79	-20.834878, -40.625399
Praia Coqueiro	Anchieta	80	-20.809891, -40.661537
Praia Ubú	Anchieta	81	-20.801528, -40.591088
Praia do Além	Anchieta	82	-20.799024, -40.583184
Praia Guanabara	Anchieta	83	-20.803165, -40.603936
Praia Inhaúma	Anchieta	84	-20.822934, -40.682831
Praia Maímba	Anchieta	85	-20.764987, -40.571244
Praia Marvila	Anchieta	86	-20.812987, -40.666016
Praia Parati	Anchieta	87	-20.804583, -40.603154
Cine - Teatro Shopping Porto de São Mateus	São Mateus	88	-18.714416, -39.854886
Museu Aberto das Tartarugas Marinhas	São Mateus	89	-18.742831, -39.747700
Museu da Imagem e Som	São Mateus	90	-18.711294, -39.853715
Museu de Arte Eclesiástica Religiosa	São Mateus	91	-18.716287, -39.866237
Praia Barra Nova	São Mateus	92	-18.975176, -39.740165
Praia do Bosque	São Mateus	93	-19.282125, -39.966893
Praia Guriri	São Mateus	94	-18.741243, -39.747498
Praia Coqueiral	Santa Cruz	95	-19.945297, -40.149902
Praia do Sauê	Santa Cruz	96	-19.923714, -40.117019
Apiário Florin	Domingos Martins	97	-20.352337, -41.047002
Casa Cultural Domingos Martins	Domingos Martins	98	-20.360792, -40.660754
Cascata do Galo	Domingos Martins	99	-20.289099, -40.649798
Fjordland Cavalgada Ecológica Pedra Azul	Domingos Martins	100	-20.396110, -41.030605
Parque Estadual Pedra Azul	Domingos Martins	101	-20.388854, -41.015415
Sítio das Flores	Domingos Martins	102	-20.392356, -41.000243
Morro do Filleti	Venda Nova do Imigrante	103	-20.326352, -41.144730
Sítio Bella Toza	Venda Nova do Imigrante	104	-20.416874, -41.080233
Praia Coqueiros	Itaúnas	105	-18.251904, -39.640299
Praia Costa Dourada	Itaúnas	106	-18.259512, -39.643631
Praia Gesuel	Itaúnas	107	-18.174560, -39.606490
Praia Itaúnas	Itaúnas	108	-18.413314, -39.697421
Praia Riacho Doce	Itaúnas	109	-18.353107, -39.673162

Casarão da Fazendo do Conhecimento	Castelo	110	-20.594064, -41.205409
Gruta do Limoeiro	Castelo	111	-20.593218, -41.204799
Parque Estadual do Forno Grande	Castelo	112	-20.485141, -41.169117
Monumento Cristo Redentor	Mimoso do Sul	113	-21.067439, -41.362660
Museu São Pedro de Alcântara	Mimoso do Sul	114	-21.065814, -41.375204
Teatro Stênio Garcia	Mimoso do Sul	115	-21.061251, -41.360796