

HEURÍSTICAS PARA A ROTEIRIZAÇÃO E PROGRAMAÇÃO DE EMBARCAÇÕES DE APOIO À EXPLORAÇÃO DE PETRÓLEO OFF-SHORE, COMBINANDO GRASP, SIMULATED ANNEALING E VND

Dalessandro Soares Vianna

Universidade Federal Fluminense, Pólo Universitário Rio das Ostras
Rua Recife, s/n – Jardim Bela Vista – CEP: 28890-000 – Rio das Ostras, RJ – Brasil
dalessandrosoares@yahoo.com.br

Edwin Benito Mitacc Meza

Universidade Federal Fluminense, Pólo Universitário Rio das Ostras
Rua Recife, s/n – Jardim Bela Vista – CEP: 28890-000 – Rio das Ostras, RJ – Brasil
emitacc@gmail.com

Eduardo Marques

Universidade Federal Fluminense, Pólo Universitário Rio das Ostras
Rua Recife, s/n – Jardim Bela Vista – CEP: 28890-000 – Rio das Ostras, RJ – Brasil
emarques@puro.uff.br

Anibal Alberto Vilcapoma Ignacio

Universidade Federal Fluminense, Pólo Universitário Rio das Ostras
Rua Recife, s/n – Jardim Bela Vista – CEP: 28890-000 – Rio das Ostras, RJ – Brasil
anibalvilcapoma@gmail.com

RESUMO

O crescente aumento de investimentos nas atividades *offshore* de petróleo tem promovido uma verdadeira corrida contra o tempo para se desenvolver toda a cadeia produtiva do setor. Neste cenário, as atividades de apoio logístico têm desafiado aos gestores a buscarem alternativas que possam reduzir os custos operacionais e ao mesmo tempo, atender a expectativa do cliente em relação ao nível de serviço ofertado. Dentre estas atividades, a programação e construção de roteiros de uma frota de embarcações, para o transporte de suprimentos para as unidades marítimas tem sido uma das mais importantes e ao mesmo tempo complexa devido ao aumento da frota e da demanda. Neste artigo são propostas duas heurísticas baseadas nas metaheurísticas GRASP e *Simulated Annealing* para o problema de programar e elaborar os roteiros de cada embarcação da frota. A primeira utiliza, em sua etapa de busca local, o método VND. A segunda também sofreu uma alteração em sua forma padrão: uma etapa de busca local foi adicionada. Desta forma, as heurísticas desenvolvidas foram nomeadas de GRASP+VND e SA+BL. Os resultados computacionais apresentados demonstram a adequação das heurísticas propostas ao problema abordado e sinalizam que a heurística GRASP+VND se destaca em relação à SA+BL conforme as dimensões do problema aumentam.

PALAVRAS CHAVE. Roteirização & Programação, Logística, Metaheurísticas.

1. INTRODUÇÃO

Diante das recentes descobertas de acumulações gigantescas de petróleo e gás natural na extensa camada Pré-Sal, a Petrobras, após mais de meio século de pesquisas e experiências acumuladas, novamente depara-se com desafios tecnológicos, gerenciais e logísticos, proporcionais à grandeza dos reservatórios localizados nessa nova fronteira tecnológica.

Nesse contexto, as atividades *off-shore* têm sido fortemente intensificadas, recebendo um percentual significativo dos investimentos para que os objetivos estratégicos da companhia e do país possam ser alavancados.

Dentre estas atividades, a logística tem mostrado uma especial importância, dado o impacto financeiro e estratégico que exerce sobre as outras atividades. Do ponto de vista estratégico, a gestão eficiente dos processos logísticos pode se apresentar como uma vantagem competitiva, sendo a fonte desta vantagem a diferenciação da organização aos olhos dos clientes e a redução dos custos de operação.

Com a busca constante por melhorias no atendimento e aumento da eficiência (incluindo reduções de custos e de tempo de entrega), as organizações investem cada vez mais em sistemas informatizados. Nos últimos anos, os operadores logísticos atuantes no Brasil procuraram ampliar a utilização de sistemas ERP (Enterprise Resource Planning ou Sistemas Integrados de Gestão Empresarial, no Brasil) e roteirizadores (SILVA, 2010; SIKILERO, 2009; SLACK et al., 2009)

Dentro da Petrobras, vários suprimentos são transportados até as Unidades Marítimas (UM's) por meio de embarcações para que os projetos de exploração e produção de petróleo possam ser desenvolvidos.

Assim, uma importante atividade é a determinação da roteirização e da programação de atendimento destas embarcações, de forma a atender todas as solicitações de suprimentos nos prazos pré-determinados para que as atividades a bordo das UM's possam ser realizadas de forma contínua, sem que haja interrupções que gerem atrasos nos projetos e a consequente perda econômica para a empresa.

A partir deste cenário bastante complexo no qual são consideradas inúmeras variáveis de decisão na elaboração das roteirizações e das programações das embarcações, é natural o surgimento de questionamentos sobre a eficácia do atual *modus operandi* aplicado à programação da frota existente. Principalmente, se forem consideradas as recentes descobertas de reservas de petróleo que apontam para um forte crescimento da demanda, o que leva a pensar em uma solução computacional para resolver o problema de programação e roteirização das embarcações.

Cabe ressaltar, que devido à existência de várias combinações possíveis dos parâmetros em um problema de programação e roteirização, nem sempre é possível encontrar na literatura um modelo que se encaixe exatamente nas condições reais estudadas. Diante deste cenário, em (HENTZY et al., 2012) é proposto um modelo matemático que permite auxiliar ao operador na determinação das rotas e da programação de atendimento de cada uma das embarcações da frota, as quais podem estar localizadas nos portos ou cumprindo escala de atendimentos. O objetivo, neste caso, é minimizar as distâncias percorridas, otimizando a utilização da frota e reduzindo os riscos de paradas operacionais nas unidades marítimas por falta de suprimentos.

Segundo Hentzy et al. (2012), o problema abordado é um problema de otimização combinatória de alta complexidade computacional, sendo do tipo *NP-Hard*. À medida que o número de UM's e embarcações aumentam, o tempo computacional requerido pelos métodos exatos se eleva de forma exponencial.

O objetivo deste trabalho é apresentar o desenvolvimento de uma metodologia, baseada nas metaheurísticas *Simulated Annealing*, GRASP e no método VND, para a solução do modelo matemático descrito por Hentzy et al. (2012), referente à roteirização e programação de embarcações.

O artigo é organizado da seguinte maneira. Na Seção 2 é feita uma descrição do problema. Na Seção 3 é apresentado um resumo das metodologias utilizadas. Já na Seção 4 são mostradas as heurísticas propostas. Na Seção 5 são apresentados os testes computacionais, e por último a Seção 6 apresenta as conclusões e trabalho futuros.

2. DESCRIÇÃO E MODELAGEM MATEMÁTICA DO PROBLEMA

Atualmente, na exploração e produção de petróleo em campos marítimos, são desenvolvidas atividades que necessitam de equipamentos de última geração e de uma série de suprimentos, tais como: água, diesel, alimentos, tubos de perfuração e produção, cimento, granéis sólidos (baritina, bentonita e calcário) e granéis líquidos (fluidos de perfuração e completação), indispensáveis à execução dos projetos de poços marítimos (HENTZY et al., 2012). Estes insumos são transportados até as Unidades Marítimas (UM's) através de embarcações de transporte de suprimentos com características operacionais diversas (tipo, tamanho, velocidade e capacidade de carga).

A etapa de roteirização e de programação das embarcações consiste em definir para cada embarcação o seu roteiro (escala de atendimento) com as tarefas de entrega (pedidos enviados pelas UM's) que cada uma deverá cumprir. Outras restrições devem ser consideradas no momento da programação, como a capacidade de transporte de cada embarcação devido à frota heterogênea.

Ao distribuir as programações para cada embarcação (roteirização e programação), o programador busca reduzir as distâncias percorridas por esta, utilizando para isto a ferramenta computacional GIS-SUB que é um software de localização geográfica que permite identificar a posição de cada UM e também calcular as distâncias entre elas.

Segundo Hentzy (2012), este problema pode ser representado por um grafo não direcionado $G(V, A)$, onde $V = \{v_1, v_2, \dots, v_n\}$ é o conjunto de todos os vértices e $A = \{(v_i, v_j): v_i, v_j \in V\}$ é o conjunto de todas as arestas. Os vértices representam clientes ou embarcações. Cada cliente pode apresentar demandas de vários produtos simultaneamente e cada embarcação pode ofertar vários produtos simultaneamente. A cada aresta (v_i, v_j) está associada uma distância não negativa d_{ij} que representa a distância entre os vértices (clientes/embarcações).

O Problema de roteirização consiste em determinar o conjunto de rotas que deverão ser feitas pelas embarcações minimizando os custos de transporte, dado pela distância e respeitando as seguintes restrições:

- a) Satisfazer a demanda de cada cliente, mesmo que este seja fracionada entre as embarcações;
- b) A soma das quantidades programadas de cada produto a ser entregue aos clientes deve ser menor ou igual ao estoque inicial da embarcação com o roteiro programado;

Assim, no modelo matemático proposto por Hentzy (2012) foram utilizadas as seguintes variáveis e notações.

- V, P : Conjunto de todos os vértices e de todos os produtos, respectivamente;
- N, C : Subconjuntos de V que representam as embarcações e os clientes;
- DEM_{ik} e CAP_{rk} : Representam a demanda do produto k pelo cliente i e a capacidade de fornecimento do produto k pela embarcação r ;
- d_{ij} : distância entre os nós (cliente/embarcação);
- x_{rij} : Variável Binária que indica se a aresta (i, j) esta sendo utilizada pela embarcação r ($x_{rij} = 1$) ou não ($x_{rij} = 0$);

- u_{ri} : Variável Binária que indica se o cliente i esta sendo atendido pela embarcação r ($u_{ri}=1$) ou não ($u_{ri}=0$);
- y_{rij} : Fluxo em cada aresta (i, j) conduzido pela embarcação r ;
- z_{rik} : Quantidade do produto k fornecido pela embarcação r ao cliente i .

Utilizando esta notação, o problema foi formulado como um modelo de programação inteira mista:

$$MIN \sum_{r \in N} \sum_{i \in V} \sum_{j \in C} d_{ij} x_{rij} \quad (1)$$

$$s.a. \quad \sum_{i \in V} y_{rij} - \sum_{i \in C} y_{rji} = u_{rj} \quad \forall r \in N, \forall j \in C \quad (2)$$

$$y_{rij} \leq M x_{rij} \quad \forall r \in N, \forall j \in C \quad (3)$$

$$\sum_{i \in V} x_{rij} = u_{rj} \quad \forall r \in N, \forall j \in C \quad (4)$$

$$\sum_{i \in C} x_{rji} \leq u_{rj} \quad \forall r \in N, \forall j \in V \quad (5)$$

$$\sum_{r \in N} z_{rik} = DEM_{ik} \quad \forall i \in C, \forall k \in P \quad (6)$$

$$\sum_{i \in C} z_{rik} \leq CAP_{rk} \quad \forall r \in N, \forall k \in P \quad (7)$$

$$z_{rik} \leq M u_{ri} \quad \forall r \in N, \forall i \in V, \forall k \in P \quad (8)$$

$$x_{rij} \in [0, 1] \quad \forall r \in N, \forall i \in V, \forall j \in C \quad (9)$$

$$u_{ri} \in [0, 1] \quad \forall r \in N, \forall i \in V \quad (10)$$

$$y_{rij} \geq 0 \quad \forall r \in N, \forall i \in V, \forall j \in C \quad (11)$$

$$z_{rik} \geq 0 \quad \forall r \in N, \forall i \in V, \forall k \in P \quad (12)$$

O objetivo (1) é minimizar a distância percorrida pelas embarcações. A equação (2) garante que a rota é conexa, ou seja, a rota inicia na embarcação que irá atendê-la e finaliza no último cliente a ser atendido, após visitar todos os outros clientes associados à rota. A desigualdade (3) assegura que o fluxo na aresta (i, j) só poderá ocorrer se a aresta (i, j) esta sendo usada por alguma embarcação. A restrição (4) garante que, deve haver exatamente uma aresta chegando no nó j vindo da embarcação r , caso j seja atendida por r . Analogamente, a desigualdade (5) assegura que, deve haver no máximo uma aresta saindo no nó j vindo da embarcação r . A equação (6) garante que a quantidade atendida do produto k tem que ser igual à demanda do cliente pelo produto k . Já que mais de um produto pode ser transportado por uma embarcação, a restrição (7) é introduzida para que a quantidade fornecida do produto k , por uma determinada embarcação r , seja menor ou igual a sua respectiva capacidade do produto k . Devido à desigualdade (8), só poderá haver fornecimento se o cliente é atendido por uma determinada embarcação, onde M é um número muito grande. O domínio adequado das variáveis é determinado pelas restrições (9), (10), (11) e (12).

Testes realizados em (HENTZY et al., 2012) mostraram que, para instâncias maiores, as soluções ótimas do modelo são praticamente impossíveis de serem obtidas em um tempo computacional factível, se comparado com o tempo que tem o programador para construir a programação e roteirização das embarcações da frota. Para estes tipos de problemas com alta complexidade computacional, existem na literatura diversos trabalhos, que fazem uso de heurísticas em diversos problemas de roteirização (HAUGHTON, 2007; RODRIGUES, 2008; MAURI; LORENA, 2009; BELFIORE; YOSHIZAKI, 2009; PARK; KIM, 2010; BRANCHINI; ARMENTANO; LØKKETANGEN, 2010; ARANHA; MONTANÉ; VIANNA, 2012; MENDES, 2007), os quais incorporam diversas restrições operacionais como janelas de tempo, capacidade homogênea/heretogênea da frota, carga fracionada, dentre outras.

3. MODELAGEM HEURÍSTICA

As duas heurísticas propostas neste trabalho são baseadas nas metaheurísticas *Simulated Annealing* (KIRKPATRICK; GELATT; VECCHI, 1983; HAESER; GOMES-RUGGIERO, 2008) e GRASP (*Greedy Randomized Adaptive Search Procedure*) (FEO; RESENDE, 1995; RESENDE; RIBEIRO, 2003). Como foram avaliadas duas estruturas de vizinhança, as quais serão detalhadas nas Subseções 4.4.1 e 4.4.2, a heurística GRASP utiliza o método VND (*Variable Neighborhood Descent*) (MLADENOVIC; HANSEN, 1997) em sua etapa de busca local.

Segundo Feo e Resende (1995), o GRASP é uma estratégia composta por múltiplos inícios, usada primeiramente na otimização de funções contínuas e em otimização combinatória. Cada iteração do GRASP consiste em duas fases: construção de uma solução inicial e busca local. Na primeira fase, a construtiva, é criada uma lista restrita de candidatos (LRC) que é formada pelos elementos cuja incorporação à solução parcial leva aos menores custos incrementais. Estes elementos podem ser, por exemplo, lotes de produção, lotes de estoque, ou seja, valores para as variáveis de decisão. A introdução de elementos para quando a solução tornar-se factível, ou seja, quando a solução atender todas as restrições do problema. O GRASP é executado por um número máximo de iterações, o qual corresponde à quantidade de soluções factíveis encontradas pela fase construtiva em que foram aplicadas a busca local. Dentre estas soluções geradas é escolhida aquela de melhor valor.

Annealing é o processo utilizado para fundir um metal, onde este é aquecido a uma temperatura elevada e em seguida é resfriado lentamente, de modo que o produto final seja uma massa homogênea. O *simulated annealing* surgiu no contexto da mecânica estatística, desenvolvido por Kirkpatrick, Gelatt e Vecchi (1983). Quando utilizada em problemas de otimização combinatória, o processo de otimização é realizado por níveis, simulando os níveis de temperatura no resfriamento. Em cada nível, dada uma solução s , vários pontos na vizinhança de s são gerados e o correspondente valor da função objetivo é calculado. Cada ponto gerado é aceito ou rejeitado de acordo com certa probabilidade. Esta probabilidade de aceitação decresce de acordo com o nível do processo, ou equivalentemente, de acordo com a temperatura.

De acordo com Mladenovic e Hansen (1997), o que difere a técnica VND do método de busca local tradicional é que ao invés de utilizar uma única estrutura de vizinhança, várias estruturas de vizinhança são utilizadas. Estas vizinhanças estendidas procuram por soluções aprimorantes que estão “mais distantes” da solução atual, assim permitindo ao método escapar de ótimos locais “de baixa qualidade” com respeito a uma vizinhança menor.

Nas Subseções 4.5 e 4.6 serão apresentados os pseudocódigos das duas heurísticas propostas neste trabalho.

4. HEURÍSTICAS PROPOSTAS

Como mencionado em seções anteriores, foram propostas neste trabalho duas heurísticas, uma baseada na metaheurística GRASP e a outra na metaheurística *Simulated Annealing* (SA). A primeira utiliza, em sua etapa de busca local, o método VND. A segunda também sofreu uma alteração em sua forma padrão: uma etapa de busca local foi adicionada. Desta forma, as heurísticas desenvolvidas foram nomeadas de **GRASP+VND** e **SA+BL**.

As subseções a seguir descrevem cada etapa desenvolvida para as heurísticas **GRASP+VND** e **SA+BL**.

4.1. REPRESENTAÇÃO DA SOLUÇÃO

A estrutura principal utilizada para codificar uma solução para o problema abordado foi um vetor v de N posições; em cada posição r deste vetor é armazenado outro vetor representando a rota percorrida pelo navio r . A Figura 1 apresenta um exemplo desta codificação para $N=4$ navios e $C=6$ clientes (unidades marítimas).

1	2	3	
2	1	6	5
3	5	3	4
4	2	1	

Figura 1. Exemplo de codificação.

Na Figura 1, por exemplo, o navio 2 visitará os clientes 1, 6 e 5, nesta ordem. É importante lembrar que, no problema abordado, um cliente pode ter a sua demanda atendida por mais de um navio.

Também é utilizada uma estrutura de dados auxiliar $Atendimento_{N \times C \times P}$, a qual informa a quantidade do produto $k \in P$ demandado pelo cliente $i \in C$ que é fornecido pelo navio $r \in N$.

4.2. FUNÇÃO OBJETIVO

Uma solução é avaliada pelo somatório das distâncias percorridas por cada navio, ou seja, pelo somatório das distâncias de todas as rotas.

4.3. MÉTODO CONSTRUTIVO GULOSO ALEATORIZADO

O método construtivo insere gradativamente um cliente i na rota de um navio r . A escolha do próximo cliente i a ser atendido, qual o navio r que irá atendê-lo e a posição s da rota do navio r em que o cliente i será inserido utiliza o seguinte critério guloso: é escolhido a tupla (i, r, s) que acrescenta a menor distância ao somatório das rotas, ou seja, a que produz o menor incremento à função objetivo.

É montada então uma lista LC de tuplas candidatas (i, r, s) , as quais são tuplas onde o cliente i possui demanda de algum produto k ainda não atendida e o navio r ainda possui oferta do mesmo produto k . De LC é montada uma lista restrita de candidatas, LRC , a qual possui os elementos de LC mais promissores. Esses elementos são aqueles que apresentam valores não superiores a $t_{min} + \alpha \times (t_{max} - t_{min})$, onde t_{min} e t_{max} representam, respectivamente, o incremento da tupla mais promissora e da menos promissora de LC ; e $\alpha \in [0,1]$ é um parâmetro de entrada do algoritmo, no qual $\alpha=0$ representa um algoritmo puramente guloso e $\alpha=1$ um método completamente aleatório.

O algoritmo construtivo utilizado neste trabalho é descrito na Figura 2.

Algoritmo Construtivo (α)**Início****Enquanto** existir um cliente ainda não atendido por completo **faça****Início**Monte a lista de candidatos LC .De LC extraia a lista restrita de candidatos LRC .Escolha uma tupla $t(i,r,s)$ aleatoriamente de LRC .Insira o cliente i na posição s da rota do navio r .Atualize a estrutura auxiliar $Atendimento$ com a quantidade máxima de cada produto $k \in P$ que o navio r pode atender da demanda do cliente i pelo produto k .**Fim****Fim****Figura 2. Algoritmo construtivo guloso aleatorizado.****4.4. ESTRUTURAS DE VIZINHANÇA**

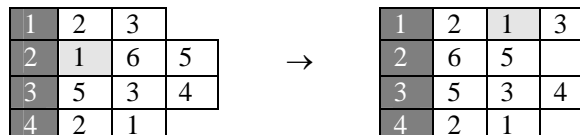
Foram utilizados dois movimentos: TROCA e REALOCAÇÃO. O primeiro é um movimento intra-rota e o segundo inter-rota. Detalhes sobre as vizinhanças obtidas através destes dois movimentos são dados nas Subseções 4.4.1 e 4.4.2.

4.4.1. Vizinhança V_TROCA

A vizinhança V_TROCA é obtida através dos movimentos TROCA. Um movimento TROCA é caracterizado pela troca de posição de dois clientes que estão na mesma rota. Ela é caracterizada pela tupla (r, i_1, s_1, i_2, s_2) , onde, na rota do navio r , o cliente i_1 que se encontra na posição s_1 passará a ocupar a posição s_2 ; e o cliente i_2 que se encontra na posição s_2 passará a ocupar a posição s_1 . A Figura 3 apresenta um exemplo de movimento TROCA, onde os clientes 1 e 5 são trocados na rota do navio 2.

**Figura 3. Exemplo de movimento TROCA.****4.4.2. Vizinhança V_REALOCAÇÃO**

A vizinhança V_REALOCAÇÃO é obtida através dos movimentos REALOCAÇÃO. Um movimento REALOCAÇÃO é caracterizado pela migração de um cliente de uma rota para outra. Ela é caracterizada pela tupla (r_1, i, s_1, r_2, s_2) , onde o cliente i que se encontra na posição s_1 da rota do navio r_1 passará a ocupar a posição s_2 da rota do navio r_2 . É importante destacar que a tupla (r_1, i, s_1, r_2, s_2) só é possível se o navio r_2 puder atender a demanda de cada produto $k \in P$ do cliente i atendida pelo navio r_1 . A Figura 4 apresenta um exemplo de movimento REALOCAÇÃO, onde o cliente 1 da rota do navio 2 (posição 1) é migrado para a rota do navio 1 (posição 2).

**Figura 4. Exemplo de movimento REALOCAÇÃO.**

Vale enfatizar uma peculiaridade deste movimento: se o cliente i já existir na rota r_2 então a rota do navio r_2 não é modificada, só alterando a estrutura $Atendimento$ para refletir este incremento no atendimento do cliente i pelo navio r_2 . A Figura 5 apresenta um exemplo deste caso, onde o cliente 1 é realocado da rota do navio 2 para a rota do navio 4.



Figura 5. Exemplo de movimento REALOCAÇÃO quando o cliente já existe na rota destino.

4.5. HEURÍSTICA GRASP+VND PROPOSTA

A Figura 6 apresenta o pseudocódigo da heurística **GRASP+VND** desenvolvida.

```

Algoritmo GRASP+VND ( $\alpha$ , tempo_limite)
Início
   $s^* \leftarrow \emptyset$ ;
  Enquanto tempo_limite não atingido faça
    Início
      Construa uma solução  $s_0$  com o algoritmo Construtivo;
      Aplique, em  $s_0$ , o método VND, utilizando as vizinhanças
        V_TROCA e V_REALOCAÇÃO gerando a solução  $s$ ;
      Se  $f(s) < f(s^*)$  então
         $s^* \leftarrow s$ ;
    Fim
  Retorne  $s^*$ ;
Fim

```

Figura 6. Heurística GRASP+VND.

No pseudocódigo da Figura 6, uma solução inicial é criada e em seguida refinada. Este processo é repetido até que o tempo limite (*tempo_limite*) de execução seja alcançado. Na etapa de construção foi usado $\alpha=0,1$. O método VND é utilizado na etapa de refinamento, o qual utiliza as vizinhanças V_TROCA e V_REALOCAÇÃO, nesta ordem. Por fim, a melhor solução encontrada é retornada.

4.6. HEURÍSTICA SA+BL PROPOSTA

A Figura 7 apresenta o algoritmo da heurística **SA+BL** desenvolvida. O algoritmo recebe como entrada a temperatura inicial T_0 , a temperatura final T_f , o número de iterações L em cada nível de temperatura, a taxa de redução de temperatura tx e o tempo de execução em segundos *tempo_limite*.

A ideia de uma heurística *Simulated Annealing* é inicialmente aceitar quase todas as transições propostas, a fim de escapar de um mínimo local de baixa qualidade (para isso, T_0 deve ser suficientemente grande) e em seguida aceitar com probabilidade cada vez menor os pontos que pioram o valor da função objetivo.

O que difere a heurística **SA+BL** de uma heurística *Simulated Annealing* padrão é a realização de uma busca local intra-rota toda vez que a solução corrente s é atualizada. Sendo assim, a vizinhança V_TROCA, apresentada na Subseção 4.4, é usada nesta etapa de refinamento, enquanto a vizinhança V_REALOCAÇÃO, também apresentada na Subseção 4.4, é usada com vizinhança padrão do *Simulated Annealing*. Para construção da solução inicial s , no início do algoritmo, foi utilizado o algoritmo **Construção**, descrito na Subseção 4.3, utilizando $\alpha=0$, o que o torna puramente guloso.


```

Algoritmo SA+BL ( $T_0, T_f, L, tx, tempo\_limite$ )
Início
 $s^* \leftarrow \emptyset$ ;
Construa uma solução  $s$  através do algoritmo Construtivo;
Enquanto  $tempo\_limite$  não atingido faça
  Início
     $T \leftarrow T_0$ ;
    Enquanto  $tempo\_limite$  não atingido e  $T \geq T_f$  faça
      Início
        Para  $w \leftarrow 1$  até  $L$  faça
          Início
            Selecione uma solução vizinha  $s'$  na vizinhança V_REALOCAÇÃO de  $s$ ;
            Se  $f(s') < f(s)$  então
               $s \leftarrow s'$ ;
              Aplique em  $s$  uma busca local usando a vizinhança V_TROCA;
            Se  $f(s) < f(s^*)$  então
               $s^* \leftarrow s$ ;
          Senão
            Seja  $b$  um valor aleatório gerado no intervalo  $[0,1)$ ;
            Se  $b < e^{-\frac{f(s)-f(s')}{T}}$  então
               $s \leftarrow s'$ ;
              Aplique em  $s$  uma busca local usando a vizinhança V_TROCA;
            Se  $f(s) < f(s^*)$  então
               $s^* \leftarrow s$ ;
          Fim
        Fim
      Fim
     $T \leftarrow T \times tx$ ;
  Fim
Retorne  $s^*$ ;
Fim

```

Figura 7. Heurística SA+BL.

5. TESTES COMPUTACIONAIS

5.1. CARACTERÍSTICAS DOS CENÁRIOS USADOS

Com o objetivo de testar a metodologia proposta foram construídos três grupos de cenários distintos para o problema programação e roteirização de embarcações. Estes grupos definem cenários de pequeno, médio e grande porte, respectivamente.

Para a construção dos dois primeiros cenários, pequeno e médio porte, foram utilizados dados obtidos de um banco de dados com informações reais de demanda, estoque, entre outras, permitindo assim uma melhor compreensão do problema abordado e suas restrições. As localizações das UM's bem como das embarcações foram obtidas através do programa desenvolvido pela equipe de Geodésia da Petrobras chamado GIS-SUB, sendo utilizadas as coordenadas geográficas UTM (N, E).

Para a construção de cenários de grande porte, foi construído um gerador de instâncias/cenários. Esse gerador recebe como entrada o número de navios, de clientes e de produtos e cria cenários onde cada atributo é definido de maneira aleatória. Foram geradas 3 instâncias com um número maior de embarcações e de UM's.

A Tabela 1 descreve as características destes tipos cenários que servirão de entrada para a realização dos testes em ambiente computacional.

Tabela 1 – Características dos cenários utilizados para testes.

Grupo	Nome do cenário	Número de Embarcações	Número de UM's (Clientes)	Número de Produtos
Pequeno Porte	PP-1	4	11	5
	PP-2	4	12	5
	PP-3	5	13	5
Médio Porte	MP-1	7	23	5
	MP-2	10	28	5
Grande Porte	GP-1	10	50	5
	GP-2	15	60	5
	GP-3	15	70	8

5.2. ANÁLISE DOS RESULTADOS

Os testes foram executados utilizando a seguinte configuração de hardware: processador Intel Core 2 Duo, memória RAM de 4.0 GB e sistema operacional Windows Vista.

Como critério de parada, para as duas heurísticas propostas, foi utilizado tempo de execução: 60 segundos para cada cenário de cada grupo.

A Tabela 2 apresenta os resultados encontrados para cenários de pequeno porte, comparando os resultados obtidos com as soluções ótimas do modelo e as melhores soluções obtidas pelas heurísticas propostas após cinco execuções. Cabe ressaltar, que devido às características dos cenários serem de pequeno porte foi possível a obtenção do valor ótimo do problema.

Tabela 2 – Resultados dos cenários de pequeno porte.

Nome	Solução Ótima (km)	SA+BL		GRASP+VND	
		(km)	Gap(%)	(km)	Gap(%)
PP-1	739	739	0,0	739	0,0
PP-2	719	719	0,0	719	0,0
PP-3	371	371	0,0	371	0,0

A Tabela 3 apresenta os resultados encontrados para cenários de médio porte, comparando os resultados obtidos com os roteiros construídos intuitivamente por decisores da empresa. No primeiro cenário, ambas heurísticas atingiram uma melhora de 45,5% em relação ao roteiro do decisor; no segundo cenário, a heurística **GRASP+VND** se sobressaiu em relação à **SA+BL**, alcançando uma melhora de 42,9% contra uma melhora de 30,4%.

Tabela 3 – Resultados dos cenários de médio porte.

Nome	Decisor (km)	SA+BL		GRASP+VND	
		(km)	Gap(%)	(km)	Gap(%)
MP-1	3711	2024	-45,5	2024	-45,5
MP-2	4881	3410	-30,4	2784	-42,9

A Tabela 4 apresenta os resultados encontrados para cenários de grande porte. Estes cenários foram montados no intuito de comparar as duas heurísticas propostas quando as dimensões do problema aumentam. A quarta coluna apresenta a diferença percentual (*Gap*) da heurística **GRASP+VND** em relação à heurística **SA+BL**. Novamente a heurística **GRASP+VND** se destacou, atingindo uma melhora média de 18,5% em relação à heurística **SA+BL**.

Tabela 4 – Resultados dos cenários de grande porte.

Nome	SA+BL (km)	GRASP+VND (km)	Gap(%)
GP-1	344	290	-15,7
GP-2	462	378	-23,2
GP-3	691	576	-16,6

6. CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi o de desenvolver heurísticas para auxiliar na programação e elaboração dos roteiros das embarcações de uma determinada frota no ambiente *off-shore*. Com esse intuito, duas heurísticas foram elaboradas, **GRASP+VND** e **SA+BL**, as quais combinam as metaheurísticas *GRASP* e *Simulated Annealing* e o método VND.

Para avaliação das heurísticas propostas, foram elaborados três grupos de testes, os quais somam oito diferentes cenários. Os dois primeiros grupos são baseados em dados reais da empresa e o último grupo foi gerado neste trabalho.

Para os cenários de pequeno porte, as heurísticas propostas alcançaram os ótimos globais divulgados em (HENTZY et al., 2012).

Para os cenários de médio porte, as heurísticas foram comparadas com os roteiros construídos intuitivamente por decisores da empresa. Ambas heurísticas foram bem superiores aos resultados descritos pelo decisor. No entanto a heurística **GRASP+VND** se sobressaiu em relação à heurística **SA+BL**.

Os cenários de grande porte foram construídos com o objetivo de avaliar as duas heurísticas propostas quando as dimensões do problema aumentam. Nestes cenários, a heurística **GRASP+VND** também se destacou em relação à **SA+BL**, obtendo um desempenho em média 18,5% superior.

Como trabalho futuro, pretende-se adicionar outras restrições ao modelo como, por exemplo, janela de tempo, coleta-entrega, limite de combustível, dentre outros.

7. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ARANHA, M. T. S.; MONTANÉ, F. A. T. & VIANNA, D. S. Heurísticas para o problema de transporte terrestre de pessoas da petrobras na Bacia de Campos. Submetido ao Brazilian Journal of Operations & Production Management, 2012.
- [2] BELFIORE, P. & YOSHIZAKI, H.T. Y. Scatter search for a real-life heterogeneous fleet vehicle routing problem with time windows and split deliveries in Brazil. European Journal of Operational Research, Vol. 199, pp. 750-758, 2009.
- [3] BRANCHINI, R. M.; ARMENTANO, V. A. & LØKKETANGEN, A. Adaptive granular local search heuristic for a dynamic vehicle routing problem. Computers & Operations Research, Vol. 36, pp. 2955-2968, 2009.
- [4] CHRISTIANSEN, M.; FAGERHOLT, K & RONEN, D. Ship routing and scheduling: status and perspectives. Transportation Science, 38:1, p. 1-18, 2004.
- [5] FEO, T. A. & RESENDE, M. G. C. Greedy randomized adaptive search procedures. Journal of Global Optimization 6, 109-133, 1995.
- [6] HAUGHTON, M. A. Assigning delivery routes to drivers under variable customer demands. Transportation Research, Part E, Vol. 43, pp. 157-172, 2007.

- [7] HAESER, G. & GOMES–RUGGIERO, M. Aspectos Teóricos de Simulated Annealing e um Algoritmo duas Fases em Otimização Global. TEMA Tend. Mat. Apl. Comput., 9, No. 3 (2008), 395-404.
- [8] HENTZY, F. C.; MEZA, E. B. M.; VIANNA, D. S. & DIANIN, M. F. V. Um modelo matemático para a programação e roteirização de embarcações de apoio à exploração de petróleo offshore. Submetido ao Encontro Nacional de Engenharia de produção, 2012.
- [9] KIRKPATRICK, S.; GELATT; C. D. & VECCHI, M. P. Optimization by simulated annealing. Science, 220 (1983), 671–680.
- [10] MAURI, G. R. & LORENA, L. A. N. Uma nova abordagem para o problema dial-a-ride, Produção, Vol. 19, No. 1, pp. 041-054, 2009.
- [11] MENDES, A.B. Programação de Frota de Apoio a Operações “Offshore” sujeita à requisição de Múltiplas Embarcações para uma mesma Tarefa. Tese de doutorado em Engenharia apresentada à Escola Politécnica da USP, 2007.
- [12] MLADENOVIC, N. & HANSEN, P. Variable Neighborhood Search. Computer and Operations Research 24, 1097-1100, 1997.
- [13] PARK, J. & KIM, B. The school bus routing problem: A review. European Journal of Operational Research, Vol. 202, pp. 311-319, 2010.
- [14] RESENDE, M. G. C. & RIBEIRO, C. C. Greedy randomized adaptive search procedures. In F. Glover e G. Kochenberger (eds.), Handbook of metaheuristics. Kluwer, 219-249, 2003.
- [15] RODRIGUES, S. B. Metaheurística Colônia de Formigas aplicada a um Problema de Roteamento de Veículos: caso da Itaipu Binacional. In: Simpósio Brasileiro de Pesquisa Operacional, 40, 2008.
- [16] SIKILERO, C. B. Contribuição das características logísticas para as prioridades competitivas: um estudo de caso em uma empresa da indústria de refrigerantes. Dissertação de Mestrado. Universidade do Vale do Rio dos Sinos - Unisinos. 2009.
- [17] SILVA, G. L. Uma nova abordagem para o problema de roteirização de veículos com restrições operacionais. Tese (Doutorado em Transportes)-Universidade de Brasília, Brasília, 2010.
- [18] SLACK, N.; CHAMBERS, S; HARRISON, A. & JOHNSTON, R. Administração da Produção. São Paulo: Atlas, 2009.