

Algoritmo Transgenético para o Passeio do Pistoneio Periódico com Realimentação Contínua nos Poços

Marco César Goldberg*, Elizabeth Ferreira Gouvêa Goldberg e Herbert de Mélo Duarte

Resumo: Este trabalho descreve um complexo problema na área de exploração de campos terrestres de petróleo, o Problema do Passeio do Pistoneio Periódico com realimentação contínua dos poços, e relata a aplicação de um algoritmo evolucionário transgenético ao problema. Os algoritmos transgenéticos se baseiam na mimetização da endossimbiose intracelular mutualista para desenvolver sua evolução artificial. São apresentados resultados de um experimento computacional com vinte e seis casos teste a fim de validar a eficiência da abordagem proposta.

Palavras-chave: Problema do passeio do pistoneio periódico, Algoritmos evolucionários, Algoritmos transgenéticos.

Abstract: *This paper describes a complex problem in the area of exploitation of oil fields, the Periodic Piston Pump Truck Tour Problem with continuous replenishment of wells, and presents the application of a transgenetic algorithm to this problem. The transgenetic algorithms are based on mimicking the intracellular mutualistic endosymbiosis to develop their artificial evolution. The results of a computational experiment on twenty-six instances are reported in order to validate the efficiency of the proposed approach.*

Keywords: *Periodic piston pump truck tour, Evolutionary algorithms, Transgenetic algorithms.*

1. Introdução

A exploração de petróleo em campos terrestres é basicamente efetuada em dois tipos de poços: poços surgentes, nos quais a pressão dos gases no interior da jazida é suficiente para fazer o óleo aflorar à superfície e poços não-surgentes, em que é necessário um processo de retirada forçada do petróleo (elevação por meios artificiais). Ao longo de sua vida útil um poço migra da condição de surgente para a de não surgente na medida em que seu reservatório vai se esgotando. Em um dado momento o decaimento da vazão do poço pode ser de tal magnitude que não mais se justifique a alocação de equipamentos fixos de elevação, indicando-se então o uso de equipamentos móveis que possam atender simultaneamente vários poços ou de outras técnicas de revitalização. No caso da opção via equipamentos móveis, uma única unidade poderá coletar o petróleo de vários poços de pequena vazão permitindo que, no intervalo entre as coletas, o reservatório tenha tempo de realimentar o poço até que alcance seu nível de equilíbrio hidrostático.

O Problema do Passeio do Pistoneio (PPP) consiste em programar o uso de um equipamento móvel de elevação de petróleo - denominado unidade móvel do pistoneio. Essa programação é traduzida em designar a seqüência de poços a serem visitados, bem como a rota que será utilizada pela unidade móvel nessa visita. O objetivo é maximizar o volume de óleo a ser extraído dentro de um horizonte de planejamento. Quando o horizonte de planejamento é suficientemente grande, os poços podem ter tempo suficiente para recompor sua oferta de petróleo após uma dada coleta sendo, portanto, passíveis de revisita. Assim, a partir de um dado horizonte de planejamento, o problema do passeio do pistoneio se transforma em um problema periódico.

O presente trabalho descreve um novo tipo de algoritmo evolucionário para a solução do Problema do Passeio do Pistoneio Periódico com Realimentação Contínua nos Poços (PPPP_R), relatando um experimento computacional associado. Na Seção 2 o Problema do Passeio do Pistoneio é descrito e é apresentada uma revisão bibliográfica dos algoritmos meta-heurísticos propostos para a solução de problemas correlatos ao presentemente abordado. Na Seção 3 é apresentado um algoritmo transgenético para a solução do problema. Na Seção 4 um experimento computacional é realizado e, finalmente, na Seção 5 conclusões são apresentadas.

*Autor para contato: marcocgold@gmail.com

2. O Problema do Passeio do Pistoneio Periódico com Realimentação Contínua

Em campos terrestres de petróleo maduros o número de poços surgentes pode ser muito pequeno. Por exemplo, no caso da Bacia Potiguar, no estado do Rio Grande do Norte, Brasil, presentemente esse número é menor que 2%. Como essa bacia possui cerca de 4.000 poços distribuídos em oito campos produtores, aproximadamente 3.800 poços serão candidatos à exploração através de métodos de elevação artificial como o *Mechanical Rod Pump*, *Progressing Cavity Pump*, bombeio centrífugo submerso, *Piston Pump*, e outros (Thomas, 2001). Contudo, em virtude de sua baixa produção, parte significativa desses poços não justifica mais o uso de equipamentos fixos de elevação sendo, portanto, candidatos ao uso de métodos de coleta móvel ou de outros métodos de recuperação avançada como *Gas Lift* e *Pig Lift*. Observa-se que o uso de técnicas de recuperação avançada são soluções clássicas no processo de incrementar o potencial de produção de campos maduros de forma economicamente viável (Lake, 2001).

Uma unidade móvel de pistoneio (UMP) consiste de um caminhão, equipado com um tanque, guincho hidráulico, cabo de aço, lança móvel e o copo de pistoneio. A UMP normalmente visita vários poços do campo a cada dia. O número dos poços visitados varia de acordo com a duração da jornada de trabalho, com o tempo gasto no percurso entre os poços, como o tempo de preparo do equipamento junto ao poço e no processo de pistoneio, em si. Ao final da jornada a UMP retorna a sua garagem.

O volume de petróleo disponível no poço por ocasião de uma visita depende, além das características do próprio poço, do tempo entre as visitas da UMP, uma vez que os poços são realimentados de óleo pelo reservatório de forma lenta e contínua assim que são esgotados pelo método de elevação artificial. O reenchimento cessa assim que a coluna de óleo do poço encontra seu equilíbrio hidrostático com o reservatório, de forma que longos períodos de repouso não aumentam a disponibilidade de óleo no poço. A realimentação dos poços obedece à função descrita na equação 1

$$\Delta V = V_{max}(1 - e^{-\frac{t}{\tau}}) \quad (1)$$

onde ΔV representa a variação no volume disponível em um poço cujo volume de óleo máximo (coluna estática) é V_{max} e que foi explotado pela última vez a t unidades de tempo, com a velocidade de re-enchimento definida pelo parâmetro τ , uma característica do poço. A variação no volume disponível no poço considera que o conteúdo do mesmo é esgotado a cada visita da UMP.

Considera-se o grafo $G = (V, A)$, onde o conjunto $V = \{0, \dots, n\}$ representa os n poços que deverão ser visitados no campo e mais a garagem da unidade móvel, representada pelo vértice 0, e $A = \{1, \dots, m\}$ o conjunto das ligações existentes no campo. O problema do Passeio do Pistoneio Periódico com Realimentação Contínua (PPPP_R) consiste em determinar um conjunto de k rotas em no grafo G , cada uma iniciando e terminando em v_0 e respeitando o limite do turno de trabalho da UMP, de forma que o óleo coletado no conjunto de k dias seja máximo. Nesse problema os poços de petróleo podem ser explotados em qualquer ocasião que se deseje, sendo que, tanto o volume disponível como o tempo de exploração necessário dependem do estágio de recuperação do poço. Presentemente considera-se que a UMP dispõe, quando necessário, de um caminhão-tanque auxiliar para o escoamento do petróleo retirado, o que representa as condições reais de trabalho da bacia Potiguar. Assim, a capacidade de retirada de óleo da UMP é considerada ilimitada.

A possibilidade de re-visita aos poços dentro de um dado horizonte de planejamento e a independência das rotas diárias estabelece o caráter periódico deste problema. Inclusive, de uma forma geral, um poço pode ser explorado antes de atingir o seu nível estático desde que o volume do poço seja atrativo em certa rota da UMP. O fato de haver realimentação dos poços após cada visita condiciona a futura oferta de petróleo ao próprio desenvolvimento da solução, o que confere ao problema um aspecto de dinamicidade determinística.

2.1 Revisão da literatura

O presente item visa resumir as pesquisas já desenvolvidas para o presente problema.

Problemas da literatura com pontos em comum com o PPPP_R

O problema do passeio do pistoneio periódico pertence à classe dos problemas de roteamento seletivo (PRVS) por não exigir que todos os vértices de demanda sejam atendidos pelo veículo de coleta e buscar a maximização do óleo retirado nos vértices. Todavia o problema possui características próprias de modo a constituir uma variante que se distingue dos problemas classicamente definidos na literatura.

A classe geral dos problemas de coleta que se assemelham ao PPPP é conhecida na literatura sob o nome de *team orienteering problem* - OP (Chao et al., 1996). O OP é geralmente definido em termos da determinação de um caminho entre específicos pontos de um dado circuito. Dessa forma diferencia-se ligeiramente do PPPP, todavia em muitas aplicações os dois pontos do caminho coincidem formando ciclo. Outros modelos que também poderiam representar o problema seriam:

- *The selective travelling salesperson problem* (Laporte & Martello, 1990). No caso do PPPP o comprimento do tour do *selective travelling salesperson* é limitado pelo tempo necessário ao deslocamento na rota e atividades nos vértices e objetivando-se maximizar o retorno nos vértices.
- *The maximum collection problem* (Kataoka & Morito, 1988; Butt & Cavalier, 1994), seria adaptável ao PPPP permitindo-se a revisita dos vértices.
- *The bank robber problem* (Balas, 1989; Arkin et al., 1998) ou outras variantes como o *Tourist Trip Design Problem* (Vansteenwegen & Oudheusden, 2007).

O problema de roteamento de veículos (PRV), de forma geral, é NP-Difícil (Backer & Furton, 1997). O problema de roteamento periódico (PRVP) é uma generalização do (PRV), e o problema de roteamento seletivo também é NP-Difícil (Golden et al., 1987).

Recentemente um procedimento baseado na meta-heurística *Scatter Search* foi desenvolvido por (Alegre et al., 2007) para a solução do problema periódico de recolhimento de sobra de materiais em uma manufatura de peças para carros. Os autores utilizam uma abordagem em duas fases. A primeira designando os pedidos aos dias e uma segunda fase construindo as rotas diárias. Hemmelmayr et al. (2009) reportam um procedimento VNS (*Variable Neighborhood Search*) para o problema.

Para o problema do roteamento seletivo de veículos (PRVS), Hayari et al. (2003) são os primeiros a proporem um algoritmo evolucionário para a versão com janelas de tempo. Tang & Miller-Hooks (2005) reportam um algoritmo de Busca Tabu. Khemakhem et al. (2005) apresentaram uma meta-heurística híbrida para a solução do PRVS, na qual uma Busca Tabu foi combinada com uma busca em vizinhança variável descendente (VND). Archetti et al. (2007) propuseram duas variantes de um algoritmo baseado em Busca Tabu e VNS. Os resultados são comparados com os resultados obtidos pelas heurísticas de Chao et al. (1996) e de Tang & Miller-Hooks (2005). Recentemente, Ke et al. (1988) reportam um algoritmo em Colônia de Formigas e Bouly et al. (2008) relatam um Algoritmo Memético.

Trabalhos desenvolvidos anteriormente em variantes do tema

Em relação ao problema de aplicação presentemente abordado, os primeiros trabalhos relatados na literatura abordaram a solução do PPP com várias simplificações. O primeiro trabalho relatado é o de Ferreira (2001) que desenvolve um Algoritmo Genético e um Algoritmo Transgenético para a solução de uma versão do Problema do Passeio do Pistoneio em que o horizonte de planejamento é fixado em no máximo de setes dias. Tal horizonte é suficiente curto para, de forma realística, desconsiderar a revisita aos poços. Dantas (2001) relata um algoritmo Busca Tabu para o PPP. Os dois trabalhos anteriormente citados apresentam resultados para a solução de casos reais associados ao campo terrestre de Canto do Amaro, localizado em Mossoró - RN. Medeiros (2001) relata um experimento computacional envolvendo o desenvolvimento de um algoritmo Genético e um Transgenético baseado exclusivamente em troca de plasmídeos tratando casos do PPP com horizonte de planejamento de 15 dias, todavia ainda preserva a simplificação de não considerar o retorno aos poços. Nesse trabalho o desempenho dos algoritmos é verificado tanto sobre situações reais no campo de Canto do Amaro como em um conjunto de instâncias testes geradas aleatoriamente em conformidade com a metodologia que é empregada e descrita no presente trabalho. O trabalho de Medeiros (2001) envolveu testes com campos com 100, 200, 300, 400, 500 e 800 poços. Dalboni (2003) relata algoritmos evolucionários sequenciais e paralelos para o problema. Dentre eles destaca-se um algoritmo genético com busca local e mineração de dados. Neste trabalho são desconsiderados os tempos de montagem e desmontagem do equipamento para retirada do petróleo, assim como o tempo de trabalho da UMP em cada poço visitado. O tempo de percurso entre os poços é considerado igual. A possibilidade de realimentação dos poços é desconsiderada. Finalmente, o PPPP é abordado por Gonçalves (2005) que desenvolve quatro algoritmos GRASP para sua solução. Os algoritmos são testados em casos gerados aleatoriamente com horizontes de planejamento variando de 1 a 15 dias. O número de poços dos campos nos casos teste varia entre 50 a 1000. Esse último trabalho considera a possibilidade do retorno a poços já visitados, todavia somente no caso de sua recuperação plena.

3. Algoritmos Transgenéticos

Os Algoritmos Transgenéticos (ATs) pertencem à classe dos algoritmos evolucionários que apóiam sua metáfora na endossimbiose intracelular mutualista como esclarece o capítulo *Transgenética Computacional* do presente livro. O processo de evolução dos algoritmos transgenéticos é desenvolvido no contexto de uma célula hospedeira. A evolução transgenética consiste em um processo co-evolucionário de adaptação entre os endossimbiontes e seu hospedeiro (Goldberg & Goldberg, 2009). Os endossimbiontes são representados, no processo evolucionário artificial, por seus cromossomos e codificam as soluções do problema em solução. A célula hospedeira possui seu próprio material genético - DNA.

3.1 Componentes do algoritmo transgenético na solução do PPPP_R

Cromossomos dos endossimbiontes

As soluções para o PPPP_R são representadas pelos cromossomos dos endossimbiontes. Esses cromossomos são k -dimensionais tendo uma dimensão associada a cada dia do horizonte de planejamento. As rotas diárias, por sua vez, indicam a seqüência de vértices visitados a partir do vértice 0. O cromossomo k -dimensional possui a estrutura exemplificada na Figura 1, em que os números nos quadrados representam os índices dos poços visitados em cada rota diária, e o vetor vertical exibe os dias associados às rotas.

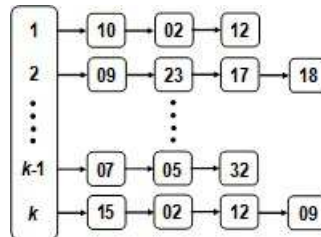


Figura 1. Representação de uma solução do PPPP.

Formação da população dos cromossomos endossimbiontes

Dois métodos de construção da solução inicial foram testados. O primeiro compreende o sorteio aleatório dos vértices de cada rota, iniciando-se e terminando-se a rota no vértice 0. O sorteio inclui novos vértices na rota até que o tempo da jornada de trabalho seja esgotado. O segundo método desenvolve um procedimento guloso aleatório semelhante à primeira fase de um algoritmo GRASP (Feo & Resende, 1995) denominado Gula. O método guloso parte da estação de tratamento de óleo inserindo sequencialmente os poços em cada rota baseado no critério de atratividade (relacionado ao volume e à distância a partir do nó atual) do poço para cada dia. Para definir a etapa de aleatorização da escolha o próximo poço a ser incluído na rota é escolhido com equiprobabilidade entre os dois poços ainda não visitados de maior atratividade, contada a rota a partir do último poço incluído na rota em formação e verificada se a inserção garante viabilidade de retorno ao vértice 0. A atratividade do poço é calculada pela expressão 2:

$$A_i^{js} = \frac{v_{is}^2}{d_{ij}} \quad (2)$$

onde A_i^{js} é a atratividade do nó i quando a UMP está no nó j e o poço i foi visitado pela última vez há s dias, v_{is} é o volume atual do poço i dado que ele foi visitado pela última vez há s dias e d_{ij} é o tempo de viagem entre os vértices i e j .

Natureza das informações contidas no hospedeiro

As informações genéticas do hospedeiro são constituídas de s subcadeias de DNA de codificação semelhante à codificação utilizada nos cromossomos dos endossimbiontes, possuindo entre duas e cinco cidades. As subcadeias de DNA possuem sua atratividade avaliada pela razão entre o volume total de óleo coletado nos poços do trecho e o tempo total utilizado na coleta. O tempo total de coleta associado a uma subcadeia de coleta é igual a soma do tempo gasto no percurso e na exploração dos vértices da subcadeia. Apenas para efeito de avaliação das cadeias considera-se a capacidade máxima em óleo dos poços coletados.

Processo de obtenção e realimentação das informações do hospedeiro

As cadeias do hospedeiro são obtidas dentre os melhores cromossomos da população de endossimbiontes. O comprimento máximo das cadeias foi determinado de forma a torná-las menores que a menor rota diária prevista nas instâncias de teste. Sempre que um cromossomo campeão é formado na população sorteia-se um dia e um vértice na rota do dia selecionado, copiando-se o trecho da rota a partir do vértice sorteado até que o comprimento definido para a cadeia seja alcançado. Formam-se r cadeias. As cadeias formadas são comparadas com as s cadeias existentes no hospedeiro preservando-se do grupo de $r + s$ cadeias as s melhores. No início do processo as informações genéticas do hospedeiro são constituídas por $s/5$ cadeias de cada um dos melhores cinco cromossomos da população de endossimbiontes.

Os vetores transgenéticos utilizados na solução do PPPP_R

O algoritmo utiliza dois tipos de vetores plasmídeos denominados de λ_1 e λ_2 . λ_1 é um plasmídeo em que a cadeia de informações é obtida diretamente do material genético do hospedeiro. A cadeia do plasmídeo λ_1

é sorteada equiprovavelmente dentre as s cadeias disponíveis no hospedeiro. λ_2 é um plasmídeo híbrido cuja cadeia de informações é obtida com a aplicação de um dentre os seguintes procedimentos heurísticos:

1. Algoritmo guloso empregado na formação da população inicial com limitação de comprimento variando entre dois e cinco poços.
2. Algoritmo guloso construtivo empregando estritamente o critério do poço mais atrativo segundo a taxa de re-enchimento. A taxa de re-enchimento pode ser obtida dividindo-se o volume total de óleo do poço cheio pelo volume que o mesmo apresenta um dia após ter sido explotado. Nesse procedimento um vértice qualquer é sorteado e ligado ao vértice 0. A partir do vértice aleatoriamente inserido a cadeia é formada de modo guloso.
3. Algoritmo que reúne poços próximos, de forma a minimizar o tempo gasto no percurso entre os poços e maximizar o aproveitamento do tempo gasto no pistoneio em si. Os poços próximos são obtidos por um processo semelhante ao da construção de uma árvore geradora mínima, mas apenas 2 a 5 poços são considerados no trecho.

O processo de ataque e transcrição dos vetores transgenéticos

Os vetores λ_1 e λ_2 empregam o mesmo operador de transcrição. A cada iteração um plasmídeo é gerado e ataca todos os cromossomos da população. A cadeia de informação dos plasmídeos é obtida do DNA do hospedeiro ou através de um dos três procedimentos construtivos. As duas fontes são sorteadas com equi-probabilidade em cada iteração do algoritmo. As fontes de informação heurísticas são submetidas à pressão de seleção natural através da contabilização do sucesso das manipulações dos plasmídeos associados. Inicialmente as fontes heurísticas recebem a mesma prioridade para a escolha, mas na medida do desenvolvimento das iterações a chance da escolha de uma heurística vai se tornar proporcional ao sucesso obtido pelos plasmídeos que dela foram derivados. O sucesso é mensurado pelo número de transcrições concretizadas. Observe-se que uma transcrição só é concretizada se melhorar a adequação do cromossomo atacado.

Durante o ataque, o plasmídeo avalia cada rota existente no cromossomo testando a possibilidade de infiltrar sua informação transportada. Uma rota será considerada sensível se não contiver algum dos poços presentes na informação do vetor. Repetidamente, cada uma das rotas sensíveis é selecionada de forma aleatória para ser manipulada pelo vetor, até onde for possível introduzir as informações da cadeia sem violar as restrições do turno de trabalho da UMP, ou até que a lista de rotas sensíveis se esgote. O plasmídeo ataca uma rota sensível tentando inserir os poços de sua cadeia de informação na seqüência em que eles se apresentam na cadeia. Os poços são examinados um a um. Um poço é simplesmente inserido se pode ser incluído entre algum par de poços existentes na rota do cromossomo preservando-se a viabilidade da rota - sua conclusão dentro do tempo máximo previsto.

Todos os poços da cadeia do plasmídeo devem ter sua inserção testada em cada uma das rotas existentes no cromossomo. Se um poço não pode ser simplesmente inserido porque sua inclusão inviabiliza a rota em todas as posições possíveis de inserção, então terá sua inserção testada considerando-se a exclusão de poços existentes na rota de forma a viabilizar o tempo de rota. Antes de qualquer inserção com exclusão em uma rota, são testadas todas as inclusões simples possíveis. Para realizar uma inserção com exclusão são removidos aleatoriamente tantos vértices quantos forem necessários para viabilizar a rota que recebeu um poço de inserção. Não existe restrição para a exclusão de vértices recentemente incluídos pela manipulação. Finalmente, a melhor solução resultante de todas as etapas do processo de ataque é mantida se for viável e melhor que a solução original.

Critérios de parada

O critério de parada adotado visa garantir a exploração do processo evolucionário do algoritmo, evitando-se a ocorrência de estagnação. A estagnação - valor η - foi fixada em 1000 iterações sem melhoria da melhor solução da população. Evitando um consumo computacional excessivo em casos particularmente difíceis, fixou-se também um número máximo total de 10.000 iterações.

Arquitetura geral do algoritmo transgenético de solução - AT_P4R

O Algoritmo 1 descreve a estratégia geral de evolução do algoritmo AT_P4R e visa auxiliar no entendimento do quadro 2 que detalha o algoritmo. Observe-se que a escolha entre a utilização do plasmídeo híbrido ou plasmídeo recombinado se dá de forma aleatória e equiprovável, como mostram os passos 4 e 5. O quadro 2 ressalta nos passos 11 e 12 que uma manipulação somente é aceita quando resulta em melhoria do cromossomo atacado e nos passos 13 e 14 esclarece que a atualização do material genético do hospedeiro se dá por ocasião da emergência de um cromossomo endossimbionte campeão.

Algoritmo 1. Algoritmo Transgenético - AT_P4R - Visão Geral

```

1.  $Pop \leftarrow iniciar\_população()$ 
2.  $GIR \leftarrow informação\_genética()$ 
3. Repita
4.    $u \leftarrow random(0,1)$ 
5.   Se  $(u \geq 0,5)$  então
6.      $\lambda \leftarrow Plasmídeo\_Híbrido(I)$ 
7.   Senão
8.      $\lambda \leftarrow Plasmídeo\_Simples(I)$ 
9.   Para todo  $C \in Pop$ 
10.     $C' \leftarrow ataque\_trans(C, \lambda)$ 
11.    Se  $C'$  é melhor que  $C$  então
12.       $C \leftarrow C'$ 
13.    Se  $C$  é um campeão então
14.       $Atualiza(GIR)$ 
15.       $j \leftarrow 0$ 
16.       $k \leftarrow k + 1; j \leftarrow j + 1$ 
17. Até  $(j > \eta$  ou  $k > \gamma)$ 

```

Arquitetura detalhada do algoritmo transgenético - AT_P4R

O Algoritmo 2 detalha o algoritmo AT_P4R. A variável *DNA* representa a informação transportada pelos plasmídeos a qual pode ser obtida diretamente das informações do hospedeiro (passo 7) ou através de três procedimentos construtivos heurísticos (passo 13). A variável heurística[*i*] contabiliza o número de transcrições concretizadas e associadas à aplicação da heurística correspondente. Os passos de 9 a 12 distribuem a probabilidade da escolha das heurísticas de formação da cadeia do plasmídeo recombinado que será utilizado para a tentativa de manipulação proporcionalmente ao número de sucessos anteriormente obtidos por essa heurística. As escolhas são inicialmente equiprováveis como mostra o passo 4. A variável *rotas* reúne as rotas do cromossomo *C* que possuem pelo menos um poço diferente dos poços que constam das rotas transportadas pelo plasmídeo. A variável *R* guarda a posição do elemento aleatório na rota de *C* que será examinada para receber a inserção de poços da cadeia do plasmídeo.

O banco de casos teste

Para validar o algoritmo foram construídas 26 instâncias para o problema a partir das instâncias do problema do caixeiro viajante do TSPLIB (Reinelt, 1991). Os casos teste foram constituídos com base em uma topologia variada e consagrada na literatura sem deixar, contudo, de serem aparelhados com dados de operação que simulassem aproximadamente os existentes em um campo real. As seguintes diretrizes foram adotadas:

- As distâncias entre os poços da instância foram transformadas em tempo de viagem entre cada par de poços. Os tempos de viagem respeitam a condição da desigualdade triangula uma vez que, em um campo real, o tempo de percurso associado à distância entre os poços - grandeza euclidiana.
- Optou-se por considerar um turno de trabalho da UMP de 16 horas (960 minutos) e um horizonte de planejamento de 30 dias.
- Os valores da produção dos poços foram gerados de forma aleatória e equiprovável dentre intervalos usualmente encontrados nos campos reais. Os valores do parâmetro da velocidade de re-enchimento - τ - foram calculados para um tempo re-enchimento de 99,9% do volume máximo dos poços através das equações de (07) a (11). Esse tempo varia aleatoriamente entre 7 e 30 dias e o volume máximo (volume da coluna estática) varia entre 4 e 12 barris.
- Os tempos de instalação e desinstalação do equipamento durante a visita a cada poço são fixados em 3 minutos cada. Adicionalmente, estabeleceu-se o tempo médio de pistoneio de um barril de óleo em 22,5 minutos. Assim o tempo de operação em um poço somente pode ser calculado conhecendo-se exatamente o dia de sua exploração e o histórico de suas explorações anteriores dentro do horizonte de planejamento.
- Da forma como foram geradas, as instâncias devem produzir, no horizonte de planejamento citado, um volume máximo de 1.280 barris de petróleo.

Algoritmo 2. Algoritmo Transgenético - AT_P4R - Visão Detalhada

```

1.  $Pop \leftarrow iniciar\_população()$ 
2.  $GIR \leftarrow informação\_genética()$ 
3.  $iterações \leftarrow 0$ ;  $sem\_melhoria \leftarrow 0$ 
4.  $heurística[1] \leftarrow n$ ;  $heurística[2] \leftarrow n$ ;  $heurística[3] \leftarrow n$ ;
    $melhorias \leftarrow 3n$ 
5. Repita
6.   Se ( $random(0,1) < 0,5$ ) então
7.      $DNA \leftarrow selecionar$  aleatoriamente informações em GIR
8.   Senão
9.      $r \leftarrow random(0, melhorias)$ 
10.    Se ( $r < heurística[1]$ ) então  $i \leftarrow 1$ 
11.    Senão Se ( $r < (heurística[1] + heurística[2])$ ) então  $i \leftarrow 2$ 
12.    Senão  $i \leftarrow 3$ 
13.     $DNA \leftarrow gerar$  informação através da heurística  $i$ 
14.  fim_se
15.  Para todo  $C \in Pop$ 
16.     $rotas \leftarrow$  conjunto de rotas de  $C$  sensíveis à manipulação pelo plasmídeo
17.    Enquanto  $rotas \neq \{\}$  e  $R'$  ainda não atribuído faça
18.       $R \leftarrow$  selecionar um elemento aleatório do conjunto  $rotas$ 
19.       $C' \leftarrow atacar(C, R, DNA)$  \ \ inserir as informações do plasmídeo em  $C$ 
        a partir da posição  $R$ 
20.      Se  $C'$  é viável então  $R' \leftarrow R$ 
21.      Senão remover elemento  $R$  do conjunto  $emprotas$ 
22.    fim_enquanto
23.    Se  $R' \neq$  nulo e  $C'$  é melhor que  $C$  então
24.      Se  $C'$  é melhor que o melhor indivíduo da população atual então
25.        inserir trechos de  $C'$  no DNA do hospedeiro
26.         $sem\_melhorias \leftarrow 0$ 
27.      fim_se
28.       $C \leftarrow C'$ ;  $heurística[i] \leftarrow heurística[i] + 1$ ;  $melhorias \leftarrow melhorias + 1$ 
29.    fim_se
30.  fim_para
31.   $iterações \leftarrow iterações + 1$ ;  $sem\_melhoria \leftarrow sem\_melhoria + 1$ 
32. Até ( $iterações > 10000$  ou  $sem\_melhoria > 1000$ )

```

Para obter o parâmetro τ (indicativo da velocidade do re-enchimento para cada poço) da equação 3.1, como citado, foi gerado aleatoriamente o tempo necessário para que o poço recuperasse 99,9% do volume máximo (entre 7 e 30 dias). Logo, τ é obtido como segue:

$$\Delta V = V_{max}(1 - e^{-\frac{t}{\tau}}) = 0,999V_{max} \quad (3)$$

$$(1 - e^{-\frac{t}{\tau}}) = 0,999 \quad (4)$$

$$1 - 0,999 = e^{-\frac{t}{\tau}} = 0,001 \quad (5)$$

$$\frac{-t}{\tau} = \ln(0,001) \quad (6)$$

$$\tau = \frac{t}{\ln(0,001)} \cong \frac{t}{6,907755} \quad (7)$$

onde t é o número de dias necessários para que o poço recupere 99,9% do volume máximo. As instâncias geradas preservam o nome do TSPLIB para caracterizar a herança da topologia de distribuição espacial dos poços.

4. Experimento Computacional

Foram realizados estudos de ajuste dos parâmetros envolvidos, a saber: a geração da população inicial, o tamanho da população de indivíduos e o tamanho do banco de informações. Em todos os testes de parâmetros, os experimentos foram realizados em um micro-computador Pentium IV 2.8GHz com 512MB de RAM. Para estes testes, foram escolhidas 8 instâncias - aproximadamente um terço do banco de testes, representativas das famílias de instâncias, e para cada valor dos parâmetros o algoritmo foi executado 10 vezes para cada uma das instâncias, tendo como critério de parada um tempo de execução de 300 segundos. Os experimentos computacionais finais do algoritmo transgenético aplicado ao PPPP_R foram realizados em um microcomputador Pentium IV 3.2GHz com 1GB de RAM. Nestes experimentos, o algoritmo foi executado 30 vezes para cada uma das 26 instâncias testadas. As instâncias possuem de 33 a 443 poços. Em todos os casos, o critério de parada adotado foi um número máximo de 10.000 iterações ou 1.000 iterações sem melhoria do melhor indivíduo da população. Os resultados obtidos estão resumidos na Tabela 1. Nela se efetua uma comparação dos resultados do algoritmo versus os obtidos com um limite superior descrito no trabalho de Duarte (2006). São apresentados o valor do limite superior, o volume máximo em número de barris, a *gap* percentual (afastamento da solução em relação ao limite superior) do volume máximo, o volume mínimo, o volume médio, o desvio-padrão (DP) do volume e o tempo médio de execução do algoritmo em segundos.

A análise dos resultados mostra uma grande estabilidade no volume do óleo coletado. Por exemplo, na solução da instância ft53, o caso que apresenta o maior *gap*, a diferença de volume entre o pior e o melhor resultado do algoritmo é de 14 barris, ou seja, uma diferença de aproximadamente 1,2%. Observe-se que, em outros casos, essa diferença é menor que 0,5% como nas instâncias ftv64 e ftv170 quando a diferença cai para 0,3% e 0,4%, respectivamente. Os resultados médios obtidos mostram que o algoritmo apresenta desvios-padrão reduzidos, nunca ultrapassando uma flutuação de mais de 5 barris ou 0,5% do total do volume. A qualidade de algumas soluções é claramente garantida, como no caso das instâncias rbg403 e 443. Coincidentemente são essas instâncias que possuem estruturas de *clusters* nos quais um grande número de poços acaba localizado em regiões restritas (afastados cerca de 1 minuto uns dos outros). Em tais instâncias é de se esperar que a rota utilize, via de regra, as menores arestas de ligação entre os poços de um *cluster*, o que significará uma razoável frequência de arestas coincidindo com as arestas consideradas no cálculo do limite inferior. Tal possibilidade minimiza o aspecto conservador do limite superior adotado, o que não acontece nos demais casos. Por outro lado, o tempo de execução mostrou-se reduzido e perfeitamente compatível com a possibilidade de futuras alterações no algoritmo, tempo normalmente em torno de 30s, ou para permitir a utilização do algoritmo em situações práticas. O tamanho das instâncias estudadas também alcança o porte dos casos reais, já que o número de poços considerados nas maiores instâncias é certamente superior ao que pode ser efetivamente designado para a operação de uma única UMP.

5. Conclusões

O problema do passeio do pistoneio periódico com realimentação contínua dos poços é um problema de otimização combinatória que apresenta elevado grau de complexidade. Pertencendo a classe dos problemas de roteamento com coleta sendo, adicionalmente, periódico e obrigado a uma estratégia de planejamento das rotas diárias que estabelece uma interdependência entre a futura oferta de óleo e o planejamento passado das rotas.

Para a solução desse problema o presente texto relata um algoritmo evolucionário da classe transgenética. Visando estabelecer uma base para a avaliação qualitativa do algoritmo o trabalho propõe um limite superior para o problema, ainda que conservativo.

O algoritmo transgenético é testado sobre um conjunto de instâncias que, mesmo preservando topologias classicamente aceita na literatura para constituir bancos de testes, busca refletir as condições da operação real em poços terrestres.

Os testes computacionais comprovam que, para efeitos práticos, o algoritmo é capaz de sugerir planejamentos de boa qualidade para campos da ordem de 400 poços designados a serem explorados exclusivamente através do pistoneio móvel e, segundo um horizonte suficientemente longo para abrigar a inclusão, em futuros trabalhos, de restrições adicionais de manutenção das unidades móveis bem como da possibilidade do uso simultâneo de mais de uma unidade móvel.

Agradecimentos

Esta pesquisa foi parcialmente suportada pelo Programa PRH-ANP 22 e pelo CNPq nos projetos 300778/2010-4 e 302819/2011-8.

Tabela 1. Resultados obtidos pelo algoritmo transgenético AT_P4R.

| Caso Teste | Limite | AT_P4R | | | | | |
|------------|---------|------------|----------|------------|--------------|------|-------|
| | | Vol Max | Gap % | Vol Min | Vol Médio | DP | T(s) |
| ft53 | 1229,80 | 1127,40 | 8,33 | 1113,50 | 1118,70 | 3,20 | 105,3 |
| ft70 | 1177,38 | 1099,70 | 6,59 | 1082,70 | 1091,40 | 3,50 | 93,30 |
| ftv100 | 1240,30 | 1168,90 | 5,75 | 1162,60 | 1165,70 | 1,40 | 35,90 |
| ftv110 | 1240,30 | 1168,50 | 5,79 | 1164,50 | 1165,90 | 0,90 | 34,30 |
| ftv120 | 1243,50 | 1176,90 | 5,36 | 1173,10 | 1175,00 | 1,00 | 37,20 |
| ftv130 | 1243,50 | 1174,90 | 5,52 | 1171,10 | 1172,50 | 1,00 | 35,20 |
| ftv140 | 1243,59 | 1179,06 | 5,19 | 1172,02 | 1175,01 | 1,44 | 32,20 |
| ftv150 | 1243,60 | 1181,20 | 5,01 | 1177,20 | 1179,40 | 0,90 | 31,40 |
| ftv160 | 1243,60 | 1178,10 | 5,27 | 1174,30 | 1176,50 | 1,00 | 38,00 |
| ftv170 | 1243,70 | 1183,10 | 4,87 | 1178,30 | 1180,60 | 1,00 | 35,70 |
| ftv33 | 1224,30 | 1129,00 | 7,79 | 1122,80 | 1125,60 | 1,40 | 26,20 |
| ftv35 | 1228,00 | 1129,90 | 7,99 | 1125,00 | 1127,50 | 1,30 | 28,20 |
| ftv38 | 1228,80 | 1140,80 | 7,16 | 1134,80 | 1137,00 | 1,40 | 27,90 |
| ftv44 | 1231,40 | 1141,00 | 7,34 | 1133,60 | 1136,10 | 1,50 | 27,20 |
| ftv47 | 1222,80 | 1128,71 | 7,69 | 1122,99 | 1125,46 | 1,47 | 29,10 |
| ftv55 | 1235,60 | 1141,98 | 7,58 | 1137,18 | 1139,30 | 1,16 | 31,30 |
| ftv64 | 1240,80 | 1167,26 | 5,93 | 1164,11 | 1165,54 | 0,98 | 29,30 |
| ftv70 | 1240,92 | 1172,97 | 5,48 | 1165,91 | 1168,86 | 1,41 | 27,10 |
| ftv90 | 1240,27 | 1166,04 | 5,98 | 1159,85 | 1162,62 | 1,71 | 33,50 |
| kro124 | 1230,87 | 1164,64 | 5,38 | 1160,87 | 1162,41 | 0,86 | 76,90 |
| p43 | 1242,64 | 1214,27 | 2,28 | 1209,62 | 1212,56 | 1,05 | 28,70 |
| rbg323 | 1178,90 | 1105,74 | 6,21 | 1095,46 | 1099,88 | 2,54 | 40,70 |
| rbg358 | 1156,46 | 1099,23 | 4,95 | 1091,48 | 1094,44 | 1,75 | 40,20 |
| rbg403 | 1140,30 | 1133,16 | 0,63 | 1126,19 | 1129,45 | 1,65 | 40,60 |
| rbg443 | 1244,27 | 1236,88 | 0,59 | 1229,14 | 1232,37 | 2,04 | 46,70 |
| ry48p | 1223,47 | 1137,67 | 7,01 | 1130,12 | 1133,71 | 1,43 | 29,30 |

Referências

- Alegre, J.; Laguna, M. & Pacheco, J., Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts. *European Journal of Operational Research*, 179(3):736–746, 2007.
- Archetti, C.; Hertz, A. & Speranza, M., Metaheuristics for the team orienteering problem. *Journal of Heuristics*, 13(1):49–76, 2007.
- Arkin, E.; Mitchell, J. & Narasimhan, G., Resource-constrained geometric network optimization. In: *Proceedings of the 14th ACM Symposium on Computational Geometry*. p. 307–316, 1998.
- Backer, B. & Furton, V., Metaheuristics in constraint programming experiments with tabu search on the vehicle routing problem. In: *Proceedings of MIC'97 2nd International Conference on Metaheuristics*. Sophia Antipolis, France, p. 1–14, 1997.
- Balas, E., The prize collecting traveling salesperson problem. *Networks*, 19(6):621–636, 1989.
- Bouly, H.; Dang, D.C. & Moukrim, A., A memetic algorithm for the team orienteering problem. In: *Applications of Evolutionary Computing*. Heidelberg: Springer, Lecture Notes in Computer Science, p. 649–658, 2008.
- Butt, S. & Cavalier, T., A heuristic for the multiple tour maximum collection problem. *Computers & Operations Research*, 21(1):101–111, 1994.
- Chao, I.M.; Golden, B. & Wasil, E.A., The team orienteering problem. *European Journal of Operational Research*, 88(3):464–474, 1996.
- Dalboni, F.L., *Algoritmos Evolutivos Eficientes para um Problema de Roteamento de Veículos*. Dissertação de mestrado, Programa de Pós-Graduação em Computação, Universidade Federal Fluminense, Niterói, RJ, 2003.
- Dantas, N.M., *Metaheurística busca tabu aplicada a solução do problema da unidade móvel do pistoneio*. Dissertação de mestrado, Programa de Pós-Graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, Natal, RN, 2001.
- Duarte, H.M., *Um estudo algorítmico de problemas logísticos na indústria de petróleo e gás natural*. Dissertação de mestrado, Programa de Pós-Graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, Natal, RN, 2006.

- Feo, T. & Resende, M., Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6(2):109–133, 1995.
- Ferreira, I., *Contaminações virais na aceleração da solução do problema do passeio do pistoneio*. Dissertação de mestrado, Programa de Pós-Graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, Natal, RN, 2001.
- Goldbarg, E.F.G. & Goldbarg, M.C., Transgenetic algorithm: A new endosymbiotic approach for evolutionary algorithms. In: Abraham, A.; Hassanien, A.E.; Siarry, P. & Engelbrecht, A. (Eds.), *Foundations of Computational Intelligence*. Heidelberg: Springer, v. 3 de *Studies in Computational Intelligence*, p. 425–460, 2009.
- Golden, B.; Levy, L. & Vohra, R., The orienteering problem. *Naval Research Logistics*, 34(3):307–318, 1987.
- Gonçalves, L.B., *Heurísticas GRASP para um Problema de Roteamento Periódico de Veículos*. Dissertação de mestrado, Programa de Pós-Graduação em Computação, Universidade Federal Fluminense, Niterói, RJ, 2005.
- Hayari, N.; Manier, M.A.; Bloch, C. & Moudni, A.E., Un algorithme évolutionniste pour le problème de tournées sélectives avec contraintes de fenêtres de temps. In: *Proceedings of 4e Conférence Francophone de Modélisation et Simulation*. Toulouse, France, 2003.
- Hemmelmayr, V.C.; Doerner, K.F. & Hartl, R.F., A variable neighborhood search heuristic for periodic routing problems. *European Journal of Operational Research*, 195(3):791–802, 2009.
- Kataoka, S. & Morito, S., An algorithm for single constraint maximum collection problem. *Journal of the Operational Research Society of Japan*, 31(4):515–530, 1988.
- Ke, L.; Archetti, C. & Feng, Z., Ants can solve the team orienteering problem. *Computers & Industrial Engineering*, 54(3):648–665, 1988.
- Khemakhem, M.; Semet, F. & Chabchoub, H., A hybrid heuristic for the selective vehicle routing problem. In: *Proceedings of MIC'05 6th International Conference on Metaheuristics*. Viena, Austria, p. 577–582, 2005.
- Lake, L.W., *Enhanced Oil Recovery*. Prentice Hall, Estados Unidos, 2001.
- Laporte, G. & Martello, S., The selective travelling salesman problem. *Discrete Applied Mathematics*, 26(2-3):193–207, 1990.
- Medeiros, A.C.C., *Algoritmo transgenético para o problema do passeio do pistoneio*. Dissertação de mestrado, Programa de Pós-Graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, Natal, RN, 2001.
- Reinelt, G., A traveling salesman problem library. *ORSA Journal on Computing*, 3(4):376–384, 1991.
- Tang, H. & Miller-Hooks, E., A tabu search heuristic for the team orienteering problem. *Computers & Operations Research*, 32(6):1379–1407, 2005.
- Thomas, J.E., *Fundamentos de Engenharia de Petróleo*. Rio de Janeiro: PETROBRAS, 2001.
- Vansteenwegen, P. & Oudheusden, D.V., The mobile tourist guide: An OR opportunity. *OR Insight*, 20(3):21–27, 2007.

Notas Biográficas

Marco César Goldbarg é Engenheiro de Fortificação e Construção (Instituto Militar de Engenharia, 1982), mestre em Sistemas e Computação (Instituto Militar de Engenharia, 1987), doutor em Sistemas e Computação pela (COPPE/UFRJ, 1990) e tem pós-doutorado em Ciência da Computação (UFMG, 1999). Atualmente é professor associado na Universidade Federal do Rio Grande do Norte.

Elizabeth Ferreira Gouvêa Goldbarg é graduada em Engenharia Industrial (Centro Federal de Educação Tecnológica – Celso Suckow da Fonseca, 1985), mestre em Sistemas e Computação (Instituto Militar de Engenharia, 1993), doutora em Engenharia de Produção (COPPE/UFRJ, 2001) e tem pós-doutorado em Sistemas e Computação (COPPE/UFRJ, 2006). Atualmente é professor associado na Universidade Federal do Rio Grande do Norte.

Herbert de Mélo Duarte é graduado em Engenharia de Computação (Universidade Federal do Rio Grande do Norte, 2004) e mestre em Sistemas e Computação (Universidade Federal do Rio Grande do Norte, 2006). Atualmente trabalha como engenheiro de software no Facebook, Inc.