

UMA “COLÔNIA DE FORMIGAS” PARA O PROBLEMA DE EXTRAÇÃO DE PETRÓLEO E OTIMIZAÇÃO DE ROTAS DE UNIDADES MÓVEIS DE PISTONEIO

DARIO JOSÉ ALOISE*, LUIZ SATORU OCHI†, DANIEL ALOISE*, ROSIERY MAIA*, VALNAIDE G. BITTENCOURT*

**DIMAP, Universidade Federal do Rio Grande do Norte (UFRN),
NATAL, RN 590-72-970, Brasil*

†*Instituto de Computação, Universidade Federal Fluminense,
Niterói, Rio de Janeiro, 24210-240, Brasil*

Emails: dario@dimap.ufrn.br, satoru@ufe.br, aloise@lcc.ufrn.br, rosiery@dimap.ufrn.br,
valnaide@engcomp.ufrn.br

Abstract— This paper focuses on the problem of oil extraction in terrestrial wells that have low productivity. A feasible strategy to maintain these wells operating is the use of a Oil Recovery Mobile System (ORS), applied at the present moment by PETROBRAS in the region of RN and CE states, Brazil. ORS is a vehicle with a tank and oil extraction mechanisms that must obey a defined daily route to visit wells in a predetermined sequence, making the complete extraction on each selected well. In this paper, a metaheuristic algorithm based on the Ant Colony Optimization (ACO) model is proposed. Computational results for real instances and randomly generated ones were used to evaluate the heuristic proposed here, and to compare it with other metaheuristics already implemented to the problem.

Resumo— Este artigo enfoca o problema de extração de Petróleo em poços terrestres que apresentam baixa produtividade. Uma estratégia viável para manter estes poços em operação é o uso de uma Unidade Móvel de Pistoneio (UMP), atualmente aplicada pela empresa PETROBRAS, na região dos estados do RN e CE, no Brasil. Uma UMP é um veículo com tanque e mecanismos de extração de óleo que deve obedecer a uma rota definida diariamente para que a mesma visite determinados poços numa seqüência dada, fazendo o pistoneio completo em cada um dos poços selecionados. Neste artigo, propomos um algoritmo metaheurístico utilizando o modelo de otimização por Colônia de Formigas. Resultados computacionais para instâncias reais e outras geradas aleatoriamente foram utilizados para avaliar a heurística aqui proposta, e para compará-la com outras metaheurísticas já implementadas para o problema.

Key Words— Exploração de Petróleo, Otimização, Metaheurística.

1 Introdução

No cenário atual de economia globalizada, um fator primordial para o desempenho de qualquer empresa, seja ela do setor público ou privado, é a racionalização dos recursos e a otimização do seu desempenho. Essa otimização torna-se fundamental à medida que as empresas estão alocadas em setores altamente competitivos, como é o caso das empresas petrolíferas. Aliado à competitividade, o desenvolvimento desse setor é primordial para a maioria dos países, onde uma significativa parcela do petróleo e seus derivados é importada, como é o caso do Brasil. Em outras palavras, quanto mais o país produz, menor será a sua parcela de importação, reduzindo com isso o déficit na sua balança comercial, como também sofrendo menor influência externa no mercado de petróleo.

A bacia potiguar de petróleo e gás, localizada nos estados do Rio Grande do Norte e Ceará (RN e CE), aparece como um ponto importante dentro desse contexto, pois representa cerca de 10% da produção nacional [segundo dados da PETROBRAS E&P, UN - RN/CE](ver [1], [2]). Nessa bacia existem alguns campos maduros, isto é al-

guns reservatórios sem energia suficiente para elevar o óleo à superfície. Nesses casos, tais reservatórios requerem instalação de métodos de elevação artificiais, tais como o bombeio mecânico cavalo-de-pau), o bombeio de cavidades progressivas, o bombeio centrífugo submerso, e o gás lift, em que se adiciona pressão ao óleo para que ele possa vencer as perdas de cargas. Contudo, em alguns casos, o custo de instalar e manter equipamentos fixos em poços são economicamente inviáveis (por exemplo, poços com pequena vazão de óleo). A Unidade Móvel de Pistoneio (UMP) é um tipo de solução viável nesses casos. Ao chegar a um desses poços, o conjunto de equipamentos da UMP é montado introduzindo no poço um copo de pistoneio suspenso por um cabo de aço, que permite a retirada do fluido por levantamento, em ações repetidas, até que se chegue ao limite de capacidade do poço, ou seja, num nível de fluido bastante baixo. Esvaziado o poço, o conjunto é desmontado e então o veículo desloca-se para o próximo poço programado na sua rota. O fluido de cada poço é armazenado no próprio tanque da UMP para posteriormente ser transferido para a estação de tratamento de óleo - ETO, que é o des-

tino de toda a produção do campo, servindo de depósito provisório.

Cada poço esvaziado será realimentado pelos furos que possui em sua base durante alguns dias, após os quais, estará novamente em condições de ser mais uma vez pistoneado. Como a capacidade do tanque de uma UMP é limitada, pode-se como alternativa, trabalhar-se com o apoio de um caminhão tanque para realizar o transporte do fluido, deixando a UMP com a tarefa exclusiva de pistonear os poços.

Embora comprovadamente válido, o uso das UMP ainda na prática padece de alguns problemas de ordem operacional, como por exemplo a incerteza no critério de definir os melhores poços que devem ser pistoneados ao longo de um dia por uma UMP. Outro problema difícil devido ao elevado número de poços, é definir para cada viagem, a rota mais eficaz (rota ótima). Estes problemas se otimizados, poderão retornar soluções de maior eficiência para a empresa, no nosso caso, a PETROBRÁS, resultando numa maximização da quantidade de petróleo ao final de cada dia, ou ao final de um dado horizonte de planejamento.

O objetivo deste artigo é propor a aplicação de algoritmos de “Colônia de Formigas” para a solução aproximada do problema aplicativo, verificando sua eficácia. A partir de agora, o problema será tratado como PCS-UMP (*Problema da Coleta Seletiva pela UMP*).

O PCS-UMP pode ser visto como uma generalização do clássico Traveling Salesman Problem (TSP), portanto é classificada como um problema NP-HARD limitando com isso o uso exclusivo de técnicas exatas para a sua solução.

Neste artigo, propomos um algoritmo metaheurístico usando o modelo de otimização por Colônia de Formigas (ACO) para o PCS-UMP. A proposta incorpora conceitos do MAX-MIN Ant System[17] e ferramentas mais eficazes na etapa de busca local. Foram efetuados testes com instâncias do PCS-UMP geradas aleatoriamente, porém com dados baseados em valores reais do campo petrolífero(ver [1],[2]). Também, um itinerário de 31 dias é gerado para um campo petrolífero real. Em instâncias de pequeno porte os resultados da Colônia de Formigas foram comparados com a solução ótima, obtida utilizando uma formulação matemática do PCS-UMP desenvolvida em [5]. Com o intuito de avaliar o desempenho em instâncias de grande porte, o algoritmo proposto é também comparado com um Algoritmo Genético e um GRASP desenvolvidos em trabalhos anteriores para o PCS-UMP[1] e [2].

O restante deste artigo é organizado da seguinte forma: a seção 2 descreve com detalhes o PCS-UMP e a seção 3 mostra uma formulação

matemática do PCS-UMP, descrito como um Problema de Programação Linear Inteira. A seção 4, apresenta o algoritmo de Colônia de Formigas proposto e a seção 5, os resultados computacionais. A análise dos resultados e conclusões são apresentadas na seção 6, seguidas pelos agradecimentos e referências bibliográficas.

2 O Problema de coleta seletiva através de UMP (PCS-UMP)

De posse de um conjunto de poços não surgentes, o problema a ser resolvido é inicialmente definir para cada dia, quais poços serão visitados (pistoneados) bem como, definir a melhor sequência de visitas aos poços selecionados. Na configuração atual da região analisada (RN/CE), existe somente uma unidade móvel de pistoneio (UMP), portanto o nosso estudo será baseado nesta suposição. Contudo os métodos propostos neste trabalho poderão facilmente serem estendidos para modelos com várias UMPs.

O PCS-UMP pode ser descrito da seguinte forma:

São dados:

i) Uma matriz das distâncias entre os poços e entre os poços e a ETO (que representa a origem de cada rota); ii) A capacidade de produção de óleo de cada poço; iii) A velocidade média de pistoneio; iv) Os tempos médios de montagem e desmontagem da UMP; v) A velocidade média de deslocamento de uma UMP na região.

As restrições do problema são:

1. Todos os poços programados para serem visitados em cada dia devem ser visitados; 2. Existe uma duração máxima da jornada diária de trabalho da UMP que deve ser respeitada; 3. A seleção de um poço para um determinado dia, deve levar em conta “o tempo de retorno de cada poço”, tempo esse que corresponde ao período mínimo necessário (geralmente alguns dias) para o seu reenchimento.

O objetivo do PCS-UMP, é gerar uma rota diária para a UMP de modo a maximizar a quantidade de óleo extraído (coletado) respeitando as restrições envolvidas.

O PCS-UMP pode ser descrito na estrutura de um grafo não direcionado $G = G(N, E, C)$ onde N representa o conjunto dos vértices (poços não surgentes); E representa o conjunto das aresta $\{i, j\}$ conectando dois poços (na prática cada aresta está associada a uma estrada ligando dois poços); e C está associado aos tempos (custos) de ir de um poço i para j ou vice-versa.

2.1 Literatura sobre o PCS - UMP

Problemas similares ao PCS-UMP são problemas de coleta e/ou despacho efetuados por um veículo de modo que apenas parte do conjunto de vértices associado é utilizado numa solução.

Dentro dessa classe de problemas, incluímos: The Prize Collecting Problem - PCP (ver [4], [7] e [9]). O PCP pode ser visto como uma generalização do TSP onde cada vértice possui associado um prêmio não negativo que será coletada se o vértice for visitado, e uma penalidade, que deverá ser paga se este vértice não for visitada. O objetivo, é gerar uma rota percorrendo um subconjunto do conjunto dos vértices, iniciando e terminando num vértice origem de modo a minimizar os custos de percurso e das penalidades pagas e coletando pelo menos um prêmio mínimo pré estabelecido.

Outro problema similar ao PCS-UMP, é denominado "The Orienteering Problem" - OP (ver [8],[13] e [15]). No OP, do mesmo modo que no PCP, para cada vértice existe um prêmio associado, mas não existe nenhuma penalidade. O objetivo no OP, é encontrar uma rota que passe por um subconjunto do conjunto de vértices, coletando a maior quantidade de prêmios possível, dentro de um limite de tempo pré-estabelecido. As demais restrições, tanto no OP como no PCP são as restrições presentes no TSP, como a de que cada vértice só deve ser visitado no máximo uma vez e que nenhum ciclo desconexo da origem deve ser considerado.

3 Uma Formulação Matemática para o PCS-UMP

Com o intuito de avaliar as heurísticas propostas neste artigo, apresentamos uma formulação matemática do PCS-UMP descrita em [3], em que o problema é descrito pelos autores como um Problema de Programação Linear Inteira:

$$\text{Maximizar } P = \sum_{i=1}^n p_i(1 - bsw_i)x_i \quad \text{s.a} \quad (1)$$

$$\sum_{i=1}^n (k_i + o_i)x_i + \sum_{i=0}^n \sum_{j=0}^n r_{ij}c_{ij} \leq L \quad (2)$$

$$\sum_{j=0}^n r_{ji} - x_i = 0 = \sum_{j=0}^n r_{ij} - x_i \quad \text{para } i = 1, \dots, n \quad (3)$$

$$\sum_{j=1}^n r_{j0} = 1 = \sum_{j=1}^n r_{0j} \quad (4)$$

$$r_{ij} \in \{0, 1\} \quad \text{para } i, j = 1, \dots, n \quad (5)$$

$$x_i \in \{0, 1\} \quad \text{para } i = 1, \dots, n \quad (6)$$

$$\sum_{(i,j) \in S} r_{ij} \leq |S| - 1 \quad S \subseteq \{1, \dots, n\} \quad (7)$$

Constantes:

n é o número total de poços disponíveis para o pistoneio;

S é o conjunto dos poços pistoneáveis;

p_i é a produção de fluidos do poço i ;

bsw_i é a taxa de água por volume recuperado do para o poço i ;

k_i é o tempo de montagem e desmontagem do equipamento no poço i ;

P é a produção total diária de óleo obtida;

o_i é o tempo de operação no poço i ;

c_{ij} menor tempo de percurso entre os poços i e j ;

L é o tamanho da jornada diária;

l é limite superior com relação à quantidade de poços que podem ser explotados respeitando o limite L .

Variáveis:

$x_i = 1$, se o poço i é explorado ($i = 1, \dots, n$), e 0 caso contrário.

$r_{ij} = 1$, se os poços i e j são consecutivos no roteamento ($i, j = 0, \dots, n$), e 0 caso contrário.

O $i = 0$ representa a ETO.

A restrição (2) garante que a quantidade de tempo gasto em uma rota não exceda a duração da jornada de trabalho L . Os tempos considerados são: os de atuação da UMP nos poços e os tempos dos percursos. Em (3) estão representadas as restrições de conservação de fluxo para os poços da rota. Em (4), se estabelece a conservação de fluxo para a ETO. Em (5) e (6) é garantido a integralidade do modelo em relação as variáveis utilizadas. Por fim, (7) é a restrição de eliminação de sub-rotas que não contém a ETO.

4 Uma "Colônia de Formigas" para o PCS-UMP

As metaheurísticas de melhor desempenho para problemas de otimização combinatória têm sido o método Tabu Search, algumas versões de Algoritmos Genéticos e Colônia de Formigas com busca local, GRASP e VNS. Dentre elas, o TS e AG são métodos já bastante explorados na literatura, o que ainda não ocorre por exemplo, com Colônia de Formigas, GRASP e VNS. No entanto apesar de serem técnicas propostas recentemente, estas têm apresentado soluções competitivas quando comparadas com as melhores versões de TS e AGs. Desta forma, para resolver aproximadamente o PCS-UMP, apresentamos um algoritmo usando o modelo de otimização ACO (*Ant Colony Optimization*), juntamente com um algoritmo de busca local. Inicialmente, apresentamos um método heurístico de construção para o PCS-UMP.

4.1 Uma heurística Construtiva para o PCS-UMP

A heurística de construção proposta para o PCS-UMP [5], é um procedimento do tipo ADD, onde a cada passo, um novo vértice é inserido a rota parcial segundo algum critério de seleção pré-definida. No caso do PCS-UMP, iniciamos com uma rota contendo somente o vértice origem (ETO), e a escolha do próximo vértice a ser inserido na rota será aquele que fornecer o maior índice de produtividade. Este índice é determinado pela razão entre o valor relativo a quantidade de óleo que seria coletado caso este vértice fosse escolhido (produção estimada (j)) e o tempo adicional gasto para incorporar este vértice a rota atual ($ta(j)$).

$$Indice(j) = \frac{\text{produção estimada}(j)}{ta(j)} \quad (8)$$

O critério de parada é o tempo limite existente para a rota do problema.

4.2 “Colônia de Formigas” Proposta

Ant Colony Optimization (ACO) [10] é uma metaheurística populacional, recentemente desenvolvida, que tem sido aplicada com sucesso em vários problemas de otimização *NP*-árduo (ver[11]). Como o próprio nome sugere, ACO é inspirado pelo comportamento de colônia de formigas reais, em particular, por seu comportamento de aquisição de alimentos. Uma das principais idéias é a comunicação indireta entre os indivíduos de uma colônia de agentes, chamados *formigas (artificiais)*, em analogia com as trilhas de feromônio que formigas reais usam para se comunicar. As trilhas de feromônio artificiais são um tipo de informação numérica distribuída, que é modificada pelas formigas para refletir sua experiência acumulada enquanto resolvendo um determinado problema.

O *Max-Min Ant System (MMAS)* [17], no qual se baseia nosso algoritmo para o PCS-UMP, é um dos algoritmos mais conhecidos e bem-sucedidos empregando o modelo ACO. O *MMAS* consegue explorar melhor o espaço de busca, permitindo que somente as formigas que obtiveram as melhores soluções adicionem feromônio durante a fase de adição destes hormônios. Também, o uso de um mecanismo simples para limitação de feromônio evita uma convergência prematura da busca no espaço de soluções. Finalmente, o *MMAS* pode facilmente ser estendido adicionando algoritmos de busca local.

Na fase de construção, cada formiga escolhe o próximo elemento baseado numa decisão probabilística. Uma formiga k , cujo último elemento inserido em sua solução é i , baseia a sua decisão do próximo elemento a ser inserido na taxa de

feromônio τ_{ij} (depositado por formigas que anteriormente escolheram a entrada do componente j na solução, partindo-se de i) e de uma informação heurística local. Portanto, componentes para a solução são escolhidos com probabilidade:

$$P_{ij}^k = \frac{[\tau(t)_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in \mathcal{N}_i^k} [\tau(t)_{il}]^\alpha \cdot [\eta_{il}]^\beta} \quad \text{se } j \in \mathcal{N}_i^k \quad (9)$$

onde α e β são dois parâmetros que determinam a importância relativa da taxa de feromônio e da informação heurística. \mathcal{N}_i^k é a vizinhança viável da formiga k , ou seja, o conjunto de componentes que a formiga k não visitou ainda.

A fase de melhorias consiste tipicamente em um procedimento de busca local, já que a solução gerada na fase de construção da Colônia de Formigas pode não representar um ótimo global. De modo geral, é sempre benéfica a aplicação de uma busca local visando a melhoria da qualidade das soluções obtidas em uma fase de construção.

4.3 Algoritmo de Construção

A fase de construção para o algoritmo de Colônia de Formigas proposto é representada pela heurística para o PCS-UMP descrita anteriormente, mas ao invés de selecionarmos, como próximo elemento (poço) da solução, o elemento j que maximiza (8), usamos a probabilidade mencionada na equação (9). O índice de produtividade de cada poço é a informação heurística local η_{ij} usada.

4.4 Atualização do Feromônio

No *MMAS* apenas uma formiga é usada para atualizar as taxas de feromônio após cada iteração do algoritmo. A regra de atualização do feromônio é dada por:

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}^{best} \quad (10)$$

onde $\Delta\tau_{ij}^{best} = \frac{f(s^{best})}{1000}$ e $f(s^{best})$ é o custo da melhor solução da iteração. Se a “melhor” formiga não escolhe pistonear o poço j a partir do poço i , $\Delta\tau_{ij}^{best} = 0$. O parâmetro ρ (com $0 \leq \rho \leq 1$) é a persistência do feromônio (portanto, $1 - \rho$ modela a evaporação). O mecanismo de evaporação ajuda a evitar uma acumulação ilimitada de feromônio, que pode levar à convergência do algoritmo para ótimos locais dentro do espaço de soluções.

4.5 Algoritmo de Busca Local

A busca local tem como objetivo refinar uma solução gerada na fase de construção da Colônia de Formigas, que muitas vezes não representa um ótimo local.

A busca local proposta em [5] utiliza técnicas já bem explorados pela literatura como os métodos ADD e DROP.

Dada uma solução inicial s^o , considere $S \subseteq N$, o subconjunto do conjunto de vértices do problema N , pertencentes a atual solução s^o e N/S , o conjunto de vértices fora da atual solução. O procedimento ADD e DROP utilizado consiste em permutar P vértices de s^o por Q vértices de N/S . O poço que aspira entrar na solução verifica previamente qual a melhor permutação possível com um poço de s^o . A troca do poço só será efetivada se o poço permutado incrementar a produção e não ultrapassar a quantidade de horas estabelecidas para a jornada diária.

5 Resultados Computacionais

O desempenho da Colônia de Formigas proposta foi comparado com outros algoritmos metaheurísticos (AG [1], GRASP[2]). Também foram feitas comparações com uma variante do método GRASP conhecida como *GRASP Reativo* (ver [6] e [16]). A fim de avaliar a performance do algoritmo proposto, a formulação matemática serviu para produzir os resultados exatos em pequenas instâncias do PCS-UMP.

São apresentados neste artigo resultados para uma instância de 200 poços e jornada de 960 minutos, com modificações dos poços disponíveis para pistoneio e da quantidade destes dentro da instância como um todo. A geração desta instância é aleatória, porém seguindo critérios reais para a geração dos dados (ver [5]). A tabela 1 exibe os resultados obtidos pelas metaheurísticas implementadas para o problema. Adicionalmente, um itinerário de trinta dias é construído a partir de uma configuração inicial do campo petrolífero *Fazenda Belém*. A produção resultante do algoritmo proposto é comparada com a produção obtida pelo itinerário efetuado pela empresa PETROBRAS durante o mesmo período (Março/1999), indicada na tabela 2. Instâncias reais são ajustadas dinamicamente a fim de respeitarem o processo de reenchimento do poço (ver [5]).

Os parâmetros utilizados para a Colônia de Formigas foram $\alpha = 1$, $\beta = 3$, $\rho = 0.95$ e listas candidatas de tamanho 10. Além disso, todos os componentes de distâncias mínimas entre dois poços são inicializadas com taxa de feromônio máxima. As taxas máximas e mínimas de feromônio são as mesmas indicadas em [17], com $P_{best} = 0.005$. Os parâmetros utilizados nos demais algoritmos metaheurísticos mencionados podem ser encontrados em [1] e [5].

Tabela 1: CS200_960

O nome das instâncias está indicado na primeira coluna. A segunda coluna mostra a quantidade de poços pistoneáveis. Exceto para 200 poços, para cada quantidade (180, 160, 140, 120 e 100) existem 5 configurações distintas de poços disponíveis. A terceira, quarta e quinta colunas apresentam os melhores resultados do AG em [1], do GRASP (G1) em [2] e do GRASP Reativo (G2), respectivamente. Os resultados médios das execuções do AG e dos métodos GRASP não foram apresentados nas referências, e portanto, não estão presentes na tabela. As demais colunas apresentam a média (AVG), produzida por 50 execuções do algoritmo, e os melhores resultados (MAX) obtidos pela Colônia de Formigas proposta. O tempo máximo permitido para a execução de cada algoritmo foi de 240s.

Inst	NP	AG	GR1	GR2	ACO	
		MAX	MAX	MAX	AVG	MAX
2-01	200	38.731	39.655	39.433	39.897	40.047
2-02	180	38.12	39.391	39.571	39.908	40.047
2-03	180	37.92	39.391	39.391	39.387	39.402
2-04	180	38	39.391	39.461	39.916	40.047
2-05	180	35.35	39.391	39.469	39.907	40.047
2-06	180	38.18	39.529	39.391	39.909	40.047
2-07	160	37.88	39.391	39.479	39.913	40.047
2-08	160	38.21	38.409	38.409	38.775	38.996
2-09	160	37.92	39.391	39.485	39.915	40.047
2-10	160	28.38	39.391	39.472	39.900	40.047
2-11	160	37.74	39.423	39.699	39.930	40.047
2-12	140	38.96	39.295	39.517	39.916	40.047
2-13	140	38.18	38.487	38.487	38.487	38.487
2-14	140	39.06	39.391	39.614	38.893	40.047
2-15	140	38.12	39.391	39.563	39.836	39.983
2-16	140	36.53	38.843	39.574	39.916	40.047
2-17	120	38.45	38.149	39.131	39.460	39.508
2-18	120	37.6	37.529	37.582	37.956	37.963
2-19	120	39.03	39.295	39.658	39.910	40.047
2-20	120	38.1	39.412	39.412	39.395	39.413
2-21	120	39.28	39.548	39.76	39.922	40.047

5.1 Análise dos resultados e Conclusões

Os resultados em destaque na tabela 1 mostram quando a Colônia de Formigas proposta produziu melhores resultados comparado às outras metaheurísticas desenvolvidas em [1] e [5]. Percebe-se que em grande parte dos resultados, a Colônia de Formigas apresenta média de soluções superiores às melhores soluções produzidas pelo GRASP e AG.

Os sub-totais e totais da tabela 2 representam a soma das produções dos dias com pistoneio registrado pela PETROBRAS em Fazenda Belém, e a produção total de óleo, respectivamente. Os valores indicam um ganho máximo de **84.5%** e de **83.4%** em média, para a produção do campo Fazenda Belém caso seguisse as seqüências de poços indicados pelas soluções da Colônia de Formigas proposta. Em [5] são ressaltados alguns pontos para as condições que geraram os resultados, com a intenção de evitar interpretações equivocadas.

O assunto deste artigo foi aplicado num projeto da PETROBRAS (Otimização do Emprego da Unidade Móvel de Pistoneio – UMP). Os resultados reais foram aplicados na empresa, na região nordeste do Brasil; mais especificamente na Bacia Potiguar Terrestre, composta por 77 poços de petróleo.

Tabela 2: Testes realizados com o campo real de Fazenda Belém

A Fazenda Belém possui 67 poços disponíveis para pistoneamento. A comparação foi feita com o mês de Março de 1999. Este mês foi selecionado, por ser aquele de maior extração para a petrobras no período de Setembro/1995 a Agosto/1999. Executou-se o algoritmo em plataforma Pentium III 700 MHz.

Dia	Petrobras		ACO	
	MAX (m^3)	MAX (m^3)	AVG (m^3)	t_{avg} (s)
1	28.16	40.198	40.171	2.46
2	7.73	37.203	37.114	3.20
3	25.13	37.182	36.907	2.60
4	36.74	35.192	34.997	3.44
5	23.26	34.406	34.248	4.40
6	4.10	34.795	34.649	4.96
7	5.86	35.190	34.985	6.08
8	16.4	33.367	33.209	4.24
9	26.07	33.567	33.093	6.84
10	5.29	33.85	33.598	4.46
11	38.65	34.904	34.788	5.72
12	28.46	32.404	32.27	5.64
13	13.67	32.752	32.748	7.08
14	12.33	33.248	33.213	3.76
15	21.8	33.771	33.524	6.22
16	25.6	32.057	31.763	4.92
17	3.73	33.118	32.734	5.94
18	19.39	32.507	32.278	5.64
19	31.44	33.431	33.015	5.70
20	15.28	33.704	33.519	4.12
21	1.67	32.582	32.332	6.22
22	25.18	32.181	32.017	5.60
23	17.25	33.299	33.136	5.60
24	23.58	33.299	33.047	6.42
25	14.9	32.063	31.89	6.56
26	14.11	32.982	32.8	5.74
27	-	34.723	34.597	5.02
28	-	32.857	32.507	5.40
29	14.05	32.548	32.475	4.36
30	36.57	32.424	32.264	4.78
31	33.08	34.925	34.587	5.58
sub-total	569.48	983.149	977.371	
total	569.48	1050.729	1044.475	

6 Agradecimentos

Esta pesquisa foi financiada por FI-NEP/CTPETRO/Brasil.

Referências

- [1] D. J. Aloise, C. A. Barros, J. A. Neves and M. Souza, "A Genetic Algorithm for a Oil Retrieval System" (ORS) (In Portuguese), Proc. of the XXXII Simpósio Brasileiro de Pesquisa Operacional (SBPO), Viçosa, MG, Brasil, 2000.
- [2] D. J. Aloise, C. A. Barros, J. A. Neves, L. S. Moura, B. W. Assmann and M. Souza, "A GRASP for a Oil Retrieval System" (ORS) (In Portuguese), Proc. of the XXXII Simpósio Brasileiro de Pesquisa Operacional (SBPO), Viçosa, MG, Brasil, 2000.
- [3] D.J.Aloise, C. A. Barros, J. A. Neves, et al. "Um algoritmo genético para uma variante do problema de orientação". Congresso Latino-Americano en Investigación de Operaciones, p4-8, Mexico, 2000.

- [4] E. Balas, "The Prize Collectin Traveling Salesman Problem", ORSA/TIMS, 1986.
- [5] C. A. Barros, "Uma aplicação de GRASP na otimização do emprego da Unidade Móvel de Pistoneio". Tese de Mestrado. Natal: UFRN, 2001.
- [6] M. P. Bastos and C. C. Ribeiro, "Reactive Tabu Search with Path-Relinking for the Steiner Problem in Graphs", In Essays and Surveys in Metaheuristics, pp. 39-58, Kluwer, 2001.
- [7] S. A. Canuto, M. G. C. Resende and C. C. Ribeiro, "Local Search with perturbations for the Prize-Collecting Steiner Tree Problem in Graphs", To appear in NETWORKS, 2001.
- [8] I. M. Chao, B. Golden and E. Wasil, "The Team Orienteering Problem" European Journal of Operational Research, vol.88, pp. 464-474, 1996.
- [9] M. Dellamico, F. Maffioli and P. Värbrand, "On Prize-Colleting Tours and the Asymmetric Traveling Salesman Problem", Dipartimento di Elettronica Comp. , Politecnico di Milano 53, 1994.
- [10] M. Dorigo and V. Maniezzo, and A. Colorni. "The Ant System: Optimization by a Colony of Cooperating Agents ". *IEEE Trans. Syst. Man Cybern. B*, 26 pages 29–41, 1996.
- [11] M. Dorigo and G. Di Caro. "The Ant Colony Optimization meta-heuristic. In D. Corne, M. Dorigo, and F. Glover, editors, *New ideas in Optimization*, pages 11–32, McGraw Hill, London, UK, 1999.
- [12] B. Golden, L. Levy and R. Vohra, "The Orienteering Problem" Naval Research Logistics, vol.34, pp. 307-318, 1987.
- [13] B. Golden, Q. Wang and L. Liu, "A multifaced heuristics for the Orienteering Problem", Naval Research Logistics, vol.35, pp. 359-366, 1988.
- [14] M. Kantor, M. Rosenwein, "The Orienteering Problem with Time Windows", Journal of Operational Research Society, vol. 43(6), pp. 629-635, 1992.
- [15] P. C. Keller, "Algorithms to solve the Orienteering Problem: A comparison", European Journal of Operational Research, vol.41, p. 224-231, 1989.

- [16] C.C.Ribeiro, E.Uchoa and R. F. Werneck, "A hybrid GRASP with perturbations and adaptive path-relinking for the Steiner Problem in Graphs", To appear in *INFORMS Journal on Computing*2001.
- [17] T. Stuetzle and H. Hoos. "MAX-MIN Ant System and Local Search for Combinatorial Optimization Problems". In *Proceedings Second International Conference on Metaheuristics, MIC'97*. Dordrecht: Kluwer Academic, 1998.
- [18] T. Tsiligirides, "Heuristics methods applied to Orienteering Problem", *Journal of the Operational Research Society*, vol.35(9), pp. 797-809, 1984.