Scheduling Workover Rigs for Onshore Oil Production

Dario J. Aloise, Daniel Aloise, Caroline T.M. Rocha

Universidade Federal do Rio Grande do Norte,
Departamento de Informática e Matemática Aplicada, Natal, RN 59072-970, Brazil
E-mails: `dario@dimap.ufrn.br, aloise@inf.puc-rio.br, crocha@ic.uff.br`

José C. Ribeiro Filho, Luiz S.S. Moura

Gerência de Elevação de Petróleo, Petrobras, UN-RNCE,
Natal, RN 59064-100, Brazil
E-mails: `jc-ribeiro@petrobras.com.br, luiz.sergio@petrobras.com.br`

Celso C. Ribeiro

Department of Computer Science, Catholic University of Rio de Janeiro,
Rio de Janeiro, RJ 22453-900, Brazil
E-mail: `celso@inf.puc-rio.br`

Abstract

Many oil wells in Brazilian onshore fields rely on artificial lift methods. Maintenance services such as cleaning, reinstatement, stimulation and others are essential to these wells. These services are performed by workover rigs, which are avaliable on a limited number with respect to the number of wells demanding service. The decision of which workover rig should be sent to perform some maintenance service is based on factors such as the well production, the current location of the workover rig in relation to the demanding well, and the type of service to be performed. The problem of scheduling workover rigs consists in finding the best schedule for the available workover rigs, so as to minimize the production loss associated with the wells awaiting for service. We propose a VNS heuristic for this problem. Computational results on synthetical and real-life problems are reported and compared with those obtained by other approaches. This project was sponsored by the Brazilian agency FINEP (*Financiadora de Estudos e Projetos*), in the framework of the CTPETRO Brazilian national plan of science and technology for oil and natural gas.

Key words: Oil production, workover rigs, VNS, heuristics, combinatorial optimization

1 Introduction

Many oil wells in Brazilian fields rely on artificial lift methods to make the oil surface. Oil can be lifted by different techniques [10], which require specialized equipment operating under dificult conditions for long periods of times. These equipments are assigned to the wells as long as their use is economically rentable. Failures of these equipments over the time require maintenance services such as cleaning, reinstatement, stimulation and others, which are essential to the exploitation of the wells. These services are performed by *workover rigs*, as illustrated in Figure 1. Workover rigs are slow mobile units moving at a speed of approximately 12 mph through a network of roads, as illustrated in Figure 2.

Fig. 1. Workover rig performing a maintenance service

Due to their high operation costs, there are relatively few workover rigs when compared with the number of wells demanding service. As an example, the state owned company Petrobras operates with eight to ten workover rigs in the Potiguar field, located in the Northeastern region of Brazil. The limited number of workover rigs may lead to service delays and inactive wells, with potentially high production loss. The decision of which workover rig should be sent to perform some maintenance service is based on factors such as the well production, the current location of the workover rigs, and the type of maintenance service to be performed.

The problem of scheduling workover rigs consists in finding the best schedule of the workover rigs to attend all wells demanding maintenance services, so as to minimize the oil production loss. The production loss of each iddle well is evaluated as its average daily flow rate under regular operation, multiplied by the number of days its production is interrupted.

Fig. 2. Transportation of a workover rig

A VNS heuristic for this problem is described in the next section. Computational results on synthetical and real-life problems are reported in Section 3 and compared with those obtained by other approaches. Concluding remarks are drawn in Section 4. This project was sponsored by the Brazilian agency FINEP (*Financiadora de Estudos e Projetos*), in the framework of the CT-PETRO Brazilian national plan of science and technology for oil and natural gas, and the results are currently under implementation at the state owned company Petrobras.

2 A VNS heuristic

In this section, we propose a Variable Neighborhood Search (VNS) heuristic for the problem of scheduling workover rigs for onshore oil production. The Variable Neighborhood Search metaheuristic proposed by Hansen and Mladenović [5, 6, 7] is based on the exploration of a dynamic neighborhood model. VNS successively explores increasing order neighborhoods in the search for improving solutions. Each iteration has two main steps: perturbation in the current neighborhood and local search. The main components of the heuristic are described next.

2.1 Initial solutions

Construction heuristics for the problem of scheduling workover rigs have been proposed and evaluated in [3, 8, 9]. Heuristic `H1` is an ADD-type procedure which will be used to build initial solutions to the VNS heuristic. Its pseudo-code is illustrated in Figure 3. We denote by R the set of wells requesting maintenance services and by S_i the ordered set of wells to be serviced by workover rig $i = 1, \dots, m$.

The schedule S_i of each workover rig $i = 1, \dots, m$ is initialized in line 1. The counter of the position *last* in which each well will be assigned is initialized in line 2. The loop in lines 3-10 is performed until all wells demanding maintenance services have been assigned to some workover rig. The loop in lines 4-8 assigns a well to the last position of each workover rig $i = 1, \dots, m$. The choice of the wells to be assigned to the workover rigs is based on their production losses. For each well $r \in R$ not yet assigned to a workover rig, we compute its production loss $loss_r(i, last)$ in case it is assigned to the last position of workover rig i. The value $loss_r(i, last)$ is equal to the estimated flow rate of well r multiplied by its iddle time once it is assigned to the last position of workover rig i. This idle time is equal to the time elapsed until the end of the maintenance of the well assigned to position $last - 1$ of workover rig i plus the traveling time this workover rig will take to reach well r plus the service time of the latter. The well r^* maximizing $loss_r(i, last)$ is selected in line 5. Next, in line 6 it is assigned to the last position of workover rig i. In line 7 it is removed from the list of wells still demanding service. Once one well has been assigned to the last position of each workover rig, the position counter *last* is increased in line 9 and a new iteration resumes. The algorithm stops when $R = \emptyset$, i.e. all wells have been assigned. Solution $S = \{S_i, i = 1, \dots, m\}$ is returned in line 11.

```
procedure H1;
1   S_i ← ∅, i = 1,...,m;
2   last ← 1;
3   while R ≠ ∅ do
4       for i = 1,...,n and R ≠ ∅ do
5           r* ← max_{r∈R}{loss_r(i, last)};
6           Insert well r* in the last position of S_i;
7           R ← R − {r*};
8       end-for;
9       last ← last + 1;
10  end-while;
11  return S = {S_i, i = 1,...,m};
end H1;
```

Fig. 3. Pseudo-code of the construction heuristic H1

2.2 Neighborhoods

We conceived nine different neighborhood definitions associated with a solution S to the problem of scheduling workover rigs. Each solution S is represented as a list of workover rigs, each of which is associated with an ordered list (defining a route and a schedule) of wells that it will service.

(1) *Swap routes* (SS): the wells and the associated routes assigned to two workover rigs are swapped, as illustrated in Figure 4 for workover rigs $S1$ and $S2$. Each solution has $m(m-1)/2$ neighbors within this neighborhood.

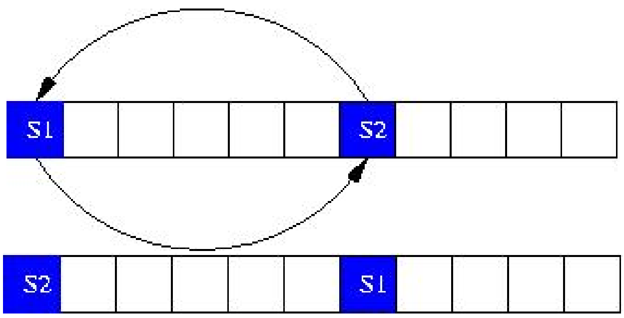

Fig. 4. Neighborhood SW

(2) *Swap wells from the same workover rig* (SWSW): the order in which two wells are serviced by the same workover rig is swapped, as illustrated in Figure 5 for wells $R2$ and $R4$ serviced by workover rig $S1$. Assuming that the n wells are evenly assigned to the m workover rigs, each solution has $n(n-m)/(2m)$ neighbors within this neighborhood.

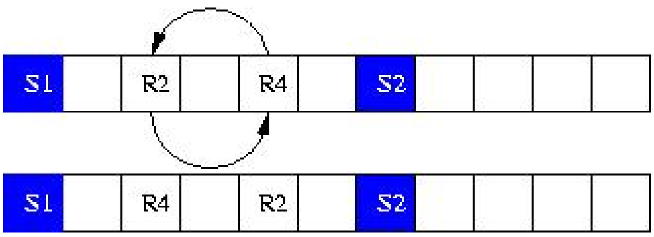

Fig. 5. Neighborhood SWSW

(3) *Swap wells from different workover rigs* (SWDW): two wells assigned to two different workover rigs are swapped, as illustrated in Figure 6 for wells $R2$ and $R8$ originally assigned respectively to workover rigs $S1$ and $S2$. Once again assuming that the n wells are evenly assigned to the m workover rigs, each solution has $n^2(m-1)/(2m)$ neighbors within this neighborhood.

(4) *Add-Drop* (AD): a well assigned to a workover rig is reassigned to any position of the schedule of another workover rig, as illustrated in Figure 7 for well $R2$ which is reassigned from workover rig $S1$ to $S2$. Once again

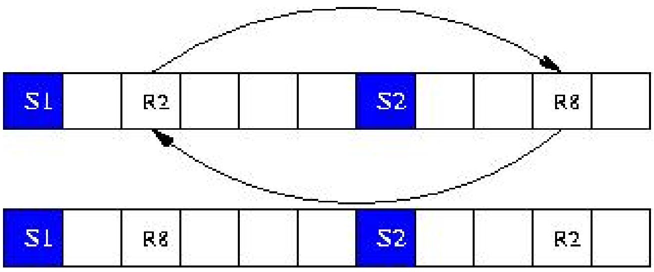

Fig. 6. Neighborhood SWDW

assuming that the n wells are evenly assigned to the m drills, each solution has also $n^2(m-1)/(2m)$ neighbors within this neighborhood.

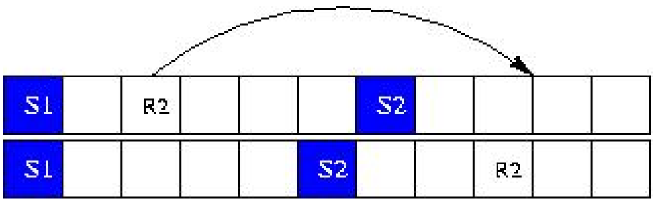

Fig. 7. Neighborhood AD

Five other neighborhoods are defined by successive applications of moves within neighborhoods SSS, SSD, and AD:

(5) SWSW^2: successively apply two moves within neighborhood SWSW
(6) SWDW^2: successively apply two moves within neighborhood SWDW
(7) SWDW^3: successively apply three moves within neighborhood SWDW
(8) AD^2: successively apply two moves within neighborhood AD
(9) AD^3: successively apply three moves within neighborhood AD

2.3 Local search

The local search procedure used at each iteration of the VNS heuristic is based on a swap neighborhood defined by all solutions which can be obtained by the exchange of a pair of wells from the current solution. This neighborhood is equivalent to the union of neighborhoods SWSW and SWDW described in the previous section.

Pairs of wells are examined in circular order. The first improving solution found is made the new current solution. The search stops at the first local optimum, after the full neighborhood of the current solution is investigated (i.e., after a sequence $n(n-1)/2$ non-improving moves are evaluated).

The nine neighborhoods described in Section 2.2 are not nested. Lower order neighborhoods are characterized by solutions which are closer to the current solution. As the neighborhood order increases, most implementations of VNS progressively investigate solutions which are farther from the current solution. Concerning the problem of scheduling workover rigs and the nine proposed neighborhoods, Add-Drop neighborhoods are the highest order ones, since many elements may change between two neighbor solutions. On the contrary, in the case of swap neighborhoods, only a few solution elements will be changed between two neighbor solutions. Our implementation of the VNS heuristic uses $k_{\max} = 9$ and investigates these neighborhoods in the following order: $N^{(1)} = \mathrm{SS}, N^{(2)} = \mathrm{SWSW}, N^{(3)} = \mathrm{SWDW}, N^{(4)} = \mathrm{SWSW}^2, N^{(5)} = \mathrm{SWDW}^2, N^{(6)} = \mathrm{SWDW}^3, N^{(7)} = \mathrm{AD}, N^{(8)} = \mathrm{AD}^2,$ and $N^{(9)} = \mathrm{AD}^3$.

procedure `VNSforWorkoverRigs`;
1 Let S be the initial solution built by `H1`;
2 $k \leftarrow 1$;
3 **while** $k \leq k_{\max}$ **do**;
4   **if** the time limit is exceeded **then return** S;
5   Randomly generate $S' \in N^{(k)}(S)$;
6   Obtain $\bar{S}$ by applying local search to S';
7   **if** $w(\bar{S}) < w(S)$ **then** $S \leftarrow \bar{S}$; $k \leftarrow 1$;
8   **else** $k \leftarrow k + 1$;
9 **end-while**;
10 Return to step 2;
end `VNSforWorkoverRigs`.

Fig. 8. Pseudo-code of the VNS heuristic for the problem of scheduling workover rigs

Figure 8 gives the algorithmic description of procedure `VNSforWorkoverRigs` which implements the VNS metaheuristic for the problem of scheduling workover rigs. A solution S and a neighborhood order k are associated with each VNS iteration. The initial soultion is built by the construction heuristic `H1` in line 1. The order k of the initial neighborhood is set to one in line 2. The loop in lines 3-9 is performed until the complete sequence $N^{(1)}, \ldots, N^{(k_{max})}$ of neighborhoods is explored. If the time limit is attained, the algorithm returns the current solution S in line 4. In line 5, a neighbor solution S' is randomly generated within neighborhood $N^{(k)}$ of solution S. Next, a solution $\bar{S}$ is obtained by applying local search to S' in line 6. If $\bar{S}$ improves the current solution, in line 7 the algorithm resumes the search from this solution using the first neighborhood. Otherwise, the algorithm resumes from S in line 8 using a higher order neighborhood. Once the complete sequence $N^{(1)}, \ldots, N^{(k_{max})}$ of neighborhoods is explored without finding any improving solution, in line 10 the algorithm returns to step 2 to reset the order of the current neighborhood to one and to resume the search from the current solution S.

3 Experimental results

In this section, we report the experimental results on synthetical and real-life problems. These results are also compared with those obtained by other approaches. We start by the description of the test problems.

3.1 Test problems

We have generated test problems according with different scenarios, based on the variation of the main parameters: daily well flow rates, distances between the wells, and service times.

(1) Daily well flow rates (in m^3/day):
- Scenario 1, in which most wells have a low-level production: 50% of the wells have a daily flow rate in the interval [3,9], 25% in the interval (9,12], and 25% in the interval (12,15].
- Scenario 2, in which most wells have a medium-level production: 25% of the wells have a daily flow rate in the interval [3,9], 50% in the interval (9,12], and 25% in the interval (12,15].
- Scenario 3, in which most wells have a high-level production: 25% of the wells have a daily flow rate in the interval [3,9], 25% in the interval (9,12], and 50% in the interval (12,15].

(2) Distances between the wells:
- Scenario 1: small distances, in which the wells are uniformly distributed in a square of 10 by 10 kilometers.
- Scenario 2: medium distances, in which the wells are uniformly distributed in a square of 20 by 20 kilometers.
- Scenario 3: large distances, in which the wells are uniformly distributed in a square of 30 by 30 kilometers.

(3) Estimated service times (in hours):
- Scenario 1, in which most wells have high service times: 50% of the wells have their service times equal to 24 hours; 25% equal to 12 hours, and 25% equal to six hours.
- Scenario 2, in which most wells have medium service times: 25% of the wells have their service times equal to 24 hours; 50% equal to 12 hours, and 25% equal to six hours.
- Scenario 3, in which most wells have small service times: 25% of the wells have their service times equal to 24 hours; 25% equal to 12 hours,

and 50% equal to six hours.

Thus, there are a total 27 different combinations of scenarios. We generated one test problem with 1500 wells for each combination. The fraction of wells demanding service is set at 10% of the total number of wells. Each synthetic test problem is identified by a string representing the associated scenarios. As an example, string `P-123` represents a test problem defined by scenario 1 for the daily well flow rates, scenario 2 for the distances between the wells, and scenario 3 for the estimated service times. All distances are Euclidean. The complete data for all test problems are available at `http://www.prometh.ufrn.br`.

3.2 Computational results

The VNS heuristic was implemented in `C`, using version 2.96 of the `gcc` compiler. The computational experiments were performed on a 1.4 GHz Pentium IV with 256 Mbytes of RAM memory and running under version 2.4.18 of the Debian implementation of Linux. The `rand` function was used for the generation of pseudo-random numbers.

Tables 1 and 2 present average results over 25 runs for problems with four workover rigs and over 20 runs for problems with eight workover rigs, each run limited at 1800 seconds of processing time. Besides the average objective function value obtained by the VNS heuristic, we also present the average results obtained by a genetic algorithm and a GRASP heuristic presented in [8] and by two ant colonies implementations (AS and MMAS) [2, 3] following the models described in [4]. The best average results among the five heuristics are shown in bold face.

Figure 9 summarizes the main results reported in Tables 1 and 2, indicating the overall number (vertical axis) and the percentage (over the bars) of problems for which each of the five heuristics obtained the best average results. The results depicted in this figure clearly indicate the dominance of the VNS heuristic with respect to the other approaches. VNS found the best average results for 85.19% of the test problems. This dominance is even clearer when one considers the best solutions found by each heuristic over all runs: in this case, the VNS heuristic always obtained the best solution.

3.3 Real-life problem

We have been provided by Petrobras with a typical real-life instance corresponding to September 5, 2002. This instance is characterized by 130 wells requiring maintenance services from nine different workover rigs.

Problem	GA	GRASP	AS	MMAS	VNS
P-111	34032.63	33347.41	30651.67	30781.03	**30359.56**
P-112	33412.64	33021.87	29801.27	29652.56	**29508.61**
P-113	31474.27	31348.80	28439.43	28453.86	**28128.51**
P-121	53874.61	53105.83	46532.29	46291.81	**46003.33**
P-122	51673.66	50637.00	45453.23	44834.11	**44788.39**
P-123	47334.47	45620.57	40398.76	39923.09	**39900.60**
P-131	80612.59	81186.06	69147.94	68824.21	**68276.79**
P-132	79119.07	76729.59	66433.99	66539.77	**65940.11**
P-133	84934.06	83704.13	68042.87	67278.61	**66582.64**
P-211	40838.59	40196.71	36978.39	36707.74	**36119.84**
P-212	38879.89	38358.96	34640.06	34634.10	**33928.53**
P-213	39523.69	39171.74	34601.73	34570.06	**34045.60**
P-221	63882.49	62133.47	55823.07	55917.56	**55496.26**
P-222	63041.73	61531.47	56608.26	**56371.71**	56496.77
P-223	63560.06	62016.97	54077.10	54155.1	**53874.62**
P-231	89651.34	87924.30	78915.01	78712.3	**78357.31**
P-232	81962.39	81002.37	73076.73	72243.76	**71690.36**
P-233	87680.83	85820.57	73598.01	72485.87	**72350.97**
P-311	42623.11	42079.79	39651.40	39786.16	**39600.5**
P-312	40926.27	40592.34	37583.19	37831.34	**37103.92**
P-313	41218.36	40938.26	38045.86	37891.44	**36954.89**
P-321	72027.49	71229.94	65001.81	64422.19	**64218.97**
P-322	69422.39	67982.10	62121.30	61692.26	**61173.41**
P-323	63561.59	63083.80	57469.00	**57170.89**	57614.69
P-331	91883.74	89209.81	81357.93	**80703.01**	80744.54
P-332	87130.90	85529.50	77878.39	77657.63	**77316.46**
P-333	87155.40	85297.10	79093.07	**78006.77**	78585.20

Table 1
Average results with four workover rigs over 25 runs of each synthetic test problem

We first compared the MMAS and the VNS heuristics on this real life instance. using the methodology proposed in [1]. Two hundred independent runs were performed for each heuristic. Each execution was terminated when a solution

Problem	GA	GRASP	AS	MMAS	VNS
P-111	16791.87	16602.51	15813.53	15815.26	**15449.50**
P-112	16516.75	16368.12	15563.90	15596.90	**15291.04**
P-113	15424.97	15288.96	14694.64	14700.65	**14262.23**
P-121	25420.56	24993.96	23720.47	23785.14	**23056.64**
P-122	25473.41	24975.59	23179.66	23174.51	**22526.75**
P-123	22598.34	22361.63	21186.87	20953.90	**20675.43**
P-131	37645.76	37117.27	34757.83	34732.97	**34116.68**
P-132	36691.23	36201.97	33527.60	33566.09	**33116.24**
P-133	37890.34	37203.06	34935.97	34957.22	**34095.64**
P-211	20016.14	19726.06	19048.13	19051.61	**18580.64**
P-212	18921.93	18615.37	17644.76	17568.49	**17205.51**
P-213	18832.29	18647.87	17615.59	17520.92	**17165.25**
P-221	30299.91	30247.54	29064.53	28978.05	**28168.54**
P-222	29976.59	29434.27	28551.89	28164.11	**27971.79**
P-223	29365.66	28841.44	27481.87	26913.17	**26639.84**
P-231	41936.89	40669.49	39405.31	**39075.96**	39367.60
P-232	36565.61	36498.49	36050.84	35524.77	**35137.29**
P-233	37404.89	37014.37	36051.76	35231.59	**34783.48**
P-311	20251.93	20094.37	19528.93	19546.10	**19434.97**
P-312	19630.09	19523.86	18906.7	18801.93	**18356.14**
P-313	20110.54	20026.43	19159.09	19081.29	**18890.91**
P-321	33097.46	32901.29	32569.79	32014.20	**31883.15**
P-322	33138.90	32696.23	31274.06	30950.60	**30830.94**
P-323	29542.89	28913.04	27770.11	**27371.39**	27666.96
P-331	43787.16	42881.27	41150.09	**40526.85**	40723.44
P-332	41462.57	41017.19	39245.86	38498.93	**38329.30**
P-333	41077.51	40970.89	39221.86	**38690.90**	39288.91

Table 2
Average results with eight workover rigs over 20 runs of each synthetic test problem

of value less than or equal to a fixed target value set at 8595.42 was found. This is a sub-optimal value chosen such that the slowest heuristic could terminate in a reasonable amount of computation time. Empirical probability

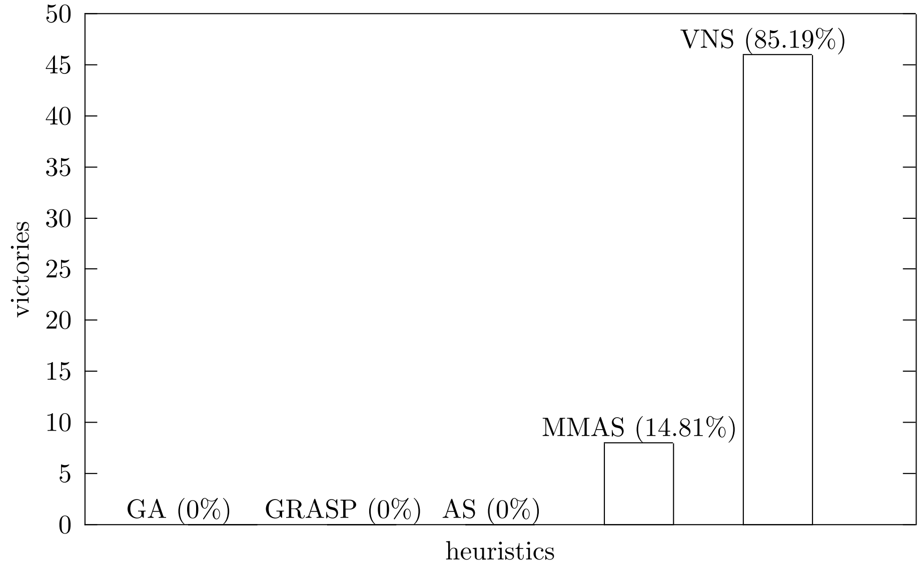

Fig. 9. Absolute and percentual number of victories of each heuristic in terms of their average results (from Tables 1 and 2)

distributions for the time-to-target-solution-value are plotted in Figure 10. To plot the empirical distribution for each algorithm, we follow the procedure described in [1]. We associate with the i-th smallest running time t_i a probability $p_i = (i - \frac{1}{2})/200$, and plot the points $z_i = (t_i, p_i)$, for $i = 1, \ldots, 200$. This figure shows that the VNS heuristic clearly outperforms the MMAS heuristic. Even though both of them find solutions with the same quality, for a given computation time, the probability of finding a solution at least as good as the target value is much higher for the VNS heuristic than for the MMAS heuristic.

So as to be able to directly compare the results obtained by the VNS heuristic with those obtained by the engineering team of Petrobras, we introduced a small modification in the problem formulation. Instead of finding the best solution such that all wells requiring maintenance services are visited, we are given a maximum time period of 15 days and we search for the best solution limited to 15 days of operation of the workover rigs.

Table 3 shows the value of the solution obtained by Petrobras and the average and best solution values over 10 runs of the MMAS ant colony heuristic and the VNS heuristic, both of them limited at 1000 seconds of processing time. The best results are indicated in bold face. We first notice that the average value of the solutions obtained by VNS is slightly better than that obtained by the MMAS ant colony implementation. The loss observed for the solution obtained by Petrobras is 2.27% greater than that obtained by the VNS heuristic. We also notice that 62 wells are serviced in the time horizon of 15 days in the solution obtained by the VNS heuristic, while only 52 were serviced in the solution obtained by MMAS and 41 in the solution produced by Petrobras.

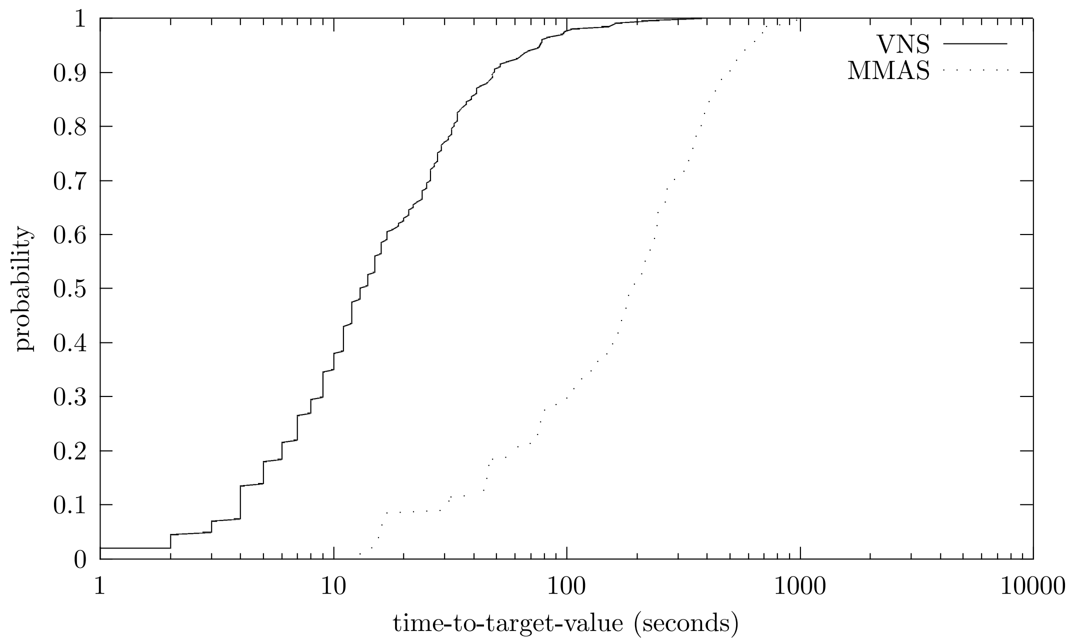

Fig. 10. Empirical of distributions of time-to-target-solution-value for MMAS and VNS

	average		best	
Heuristic	loss (m^3)	normalized loss	loss (m^3)	normalized loss
Petrobras	4919.50	2.27%	4919.50	2.27%
MMAS	4810.91	0.01%	4810.65	0.01%
VNS	4810.38	-	4810.18	-

Table 3
Results for the real-life instance for a time horizon of 15 days

4 Concluding remarks

This project was sponsored by the Brazilian agency FINEP (*Financiadora de Estudos e Projetos*), in the framework of the CTPETRO Brazilian national plan of science and technology for oil and natural gas. We now comment on the economical impact of the results obtained with the use of the approach proposed in this work.

There are usually ten workover rigs operating full time in the Potiguar field, located in the Northeastern region of Brazil. These workover rigs are subcontracted from their owners and their rental cost is approximately US$ 10,000,000 per year to Petrobras. We obtained a daily reduction of 109 m^3 (equivalent to 685.6 bbl) in the production losses along 15 days, corresponding to the difference between the solution obtained by the new heuristic and that computed by Petrobras, as depicted in Table 3. The price of the Brent oil was quoted at 27.82/bbl on June 13, 2003. If we consider the reduction in the financial losses estimated at 30% due to the increase in the number of wells serviced (21) by the new solution, which corresponds to 1463 m^3 (9,199 bbl) in the same period, we obtain savings of approximately US$ 550,000 in one month, or US$ 6,600,000 per year. These savings do not take into account labor costs and other operational costs.

These results are much better than the gains expected when this project was contracted, which were originally estimated as 5 to 10% of the yearly rental costs, i.e. US$ 500,000 to 1,000,000 per year. As a consequence, the new heuristic approach is currently under implementation at the state owned company Petrobras. The expected savings in oil production loss have opened the path to preliminary studies to investigate the gains that could be obtained if additional workover rigs were used. Further numerical results cannot be disclosed at the present time.

References

[1] R.M. Aiex, M.G.C. Resende, and C.C. Ribeiro. Probability distribution of solution time in GRASP: An experimental investigation. *Journal of Heuristics*, 8:343–373, 2002.

[2] D. Aloise. Novas abordagens metaheurísticas para o problema de otimização do emprego de sonda de produção terrestre. Graduation project, Department of Computer Science and Applied Mathematics, Federal University of Rio Grande do Norte, Brazil, 2002 (in Portuguese).

[3] D. Aloise, T.F. Noronha, R.S. Maia, V.G. Bittencourt, and D.J. Aloise. Heurísticas de colonia de formigas com path-relinking para o problema de otimização da alocação de sondas de produção terrestre. In *Proceedings of the XXXIV Brazilian Symposium on Operations Research*, Rio de Janeiro, 2002 (in Portuguese).

[4] M. Dorigo and T. Stützle. The ant colony optimization metaheuristic: Algorithms, applications, and advances. In F. Glover and G. Kochenberger, editors, *Handbook of Metaheuristics*, pages 251–285. Kluwer Academic Publishers, 2003.

[5] P. Hansen and N. Mladenović. An introduction to variable neighbourhood search. In S. Voss, S. Martello, I.H. Osman, and C. Roucairol, editors, *Metaheuristics: Advances and trends in local search procedures for optimization*, pages 433–458. Kluwer Academic Publishers, 1999.

[6] P. Hansen and N. Mladenović. Variable neighborhood search. In F. Glover and G. Kochenberger, editors, *Handbook of Metaheuristics*, pages 145–184. Kluwer Academic Publishers, 2003.

[7] N. Mladenović and P. Hansen. Variable neighbourhood search. *Computers and Operations Research*, 24:1097–1100, 1997.

[8] T.F. Noronha. Uma heurística GRASP para o itinerário de sondas de produção terrestre. Graduation project, Department of Computer Science and Applied Mathematics, Federal University of Rio Grande do Norte, Brazil, 2001 (in Portuguese).

[9] T.F. Noronha, F.G. Lima Jr., and D.J. Aloise. Um algoritmo heurístico guloso aplicado ao problema do gerenciamento das intervenções em poços petrolíferos por sondas de produção terrestre. In *Proceedings of the XXXIII Brazilian Symposium on Operations Research*, page 135, Campos do Jordão, 2001 (in Portuguese).

[10] J.E. Thomas. *Fundamentos de engenharia de petróleo.* Interciência, 1999 (in Portuguese).