

UM ALGORITMO HEURÍSTICO PARA O PROBLEMA DE GERENCIAMENTO DE ESCALA OPERACIONAL DE CONTROLADORES DE TRÁFEGO AÉREO

Bruno Mota Avelar Almeida, Túlio Ângelo Machado Toffolo,
Marcone Jamilson Freitas Souza

Universidade Federal de Ouro Preto
Departamento de Computação
Instituto de Ciências Exatas e Biológicas
sgt.avelar@gmail.com



19 de dezembro de 2011



Sumário

- 1 **Introdução**
 - Literatura
 - Descrição do Problema
 - Proposta
 - Metaheurísticas Utilizadas
- 2 **Abordagem Heurística**
 - Representação de uma Solução
 - Avaliação de uma Solução
 - Estruturas de vizinhança
- 3 **O Algoritmo MSGVNS**
 - Construção de uma Solução Inicial
 - Algoritmo MSGVNS
- 4 **Resultados**
- 5 **Conclusões**



Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo

Trabalhos da literatura

Problemas relacionados

- Escalonamento de Horários;
- Escalonamento de Salas de Aulas;
- Escalonamento de Motoristas;
- Escalonamento de Enfermeiros;
- Escalonamento de Tripulantes.



Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo

Trabalhos da literatura

Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo - PGECTA

O Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo (PGECTA) tem algumas características desses problemas apontados na literatura

A diferença entre esses problemas reside, sobretudo, nas regras trabalhistas existentes e em algumas restrições operacionais.



Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo

Descrição

Cada CTA deve contabilizar ao máximo 20 turnos ao final de cada mês, sendo que as restrições operacionais não devem ser infringidas:

- Mínimo de 12 CTAs no turno da manhã, 13 no da tarde e 9 no pernoite;
- Máximo de 10 manhãs por mês por CTA;
- Máximo de 10 tardes por mês por CTA;
- Máximo de 10 pernoites por mês por CTA;
- Máximo de 3 pernoites consecutivos por CTA;
- Máximo de 6 serviços consecutivos sem um dia inteiro de folga por CTA;
- Máximo de 6 dias consecutivos de folga por CTA;
- Período mínimo de descanso entre serviços de 1 turno por CTA.



Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo

Caracterização

PGECTA

A complexidade do problema se assenta no fato de o mesmo ter que atender, além das restrições operacionais, restrições como preferências de horários por parte dos controladores

Por exemplo:

Um CTA c não pode ou não deseja trabalhar entre os dias: 20-25 do mês e não poderá trabalhar no turno da manhã no dia 30.

Para este CTA, além das restrições descritas anteriormente, teremos um conjunto de objetivos individuais que poderão ser ou não atendidos.



Problema de Gerenciamento de Escala dos Controladores de Tráfego Aéreo

Proposta

- Algoritmo MSGVNS, baseada nos procedimentos:
 - *Multi-Start* (MS)
 - *General Variable Neighborhood Search* (GVNS)
- Vantagens: Combina o poder do método Multi-Start para gerar uma boa solução inicial e a flexibilidade do método GVNS, o qual utiliza como procedimento de busca local o método VND.



Multi-Start

Pseudocódigo Básico

Algoritmo 1: Multi-Start

Entrada: Função $f(\cdot)$, $N(\cdot)$

```
1  $f^* \leftarrow \infty$ 
2 enquanto Critério de parada não atendido faça
3    $s \leftarrow \text{ConstroiSolucao}()$ 
4    $s \leftarrow \text{BuscaLocal}()$ 
5   se  $f(s) < f(s^*)$  então
6      $s^* \leftarrow s$ 
7      $f^* \leftarrow f(s)$ 
8   fim
9 fim
10  $s \leftarrow s^*$ 
11 retorna  $s$ 
```



Multi-Start

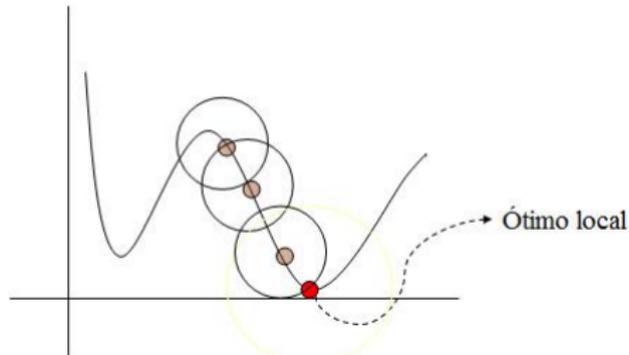
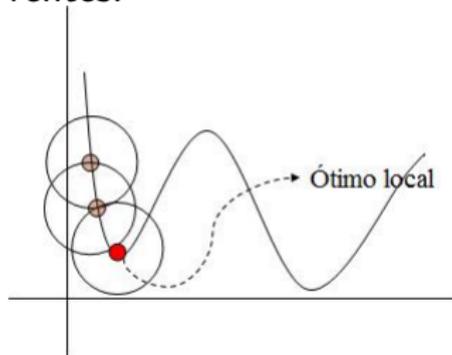
- Geração aleatória de soluções;
- Pode escapar de ótimos locais;
- Diversificação no espaço de busca;
- Fácil implementação e abstração.



GVNS

Procedimento Básico

A Busca Geral em Vizinhança Variável, ou *General Variable Neighborhood Search* (GVNS), é uma metaheurística inspirada na ideia de buscar boas soluções por meio de estruturas de vizinhança diferentes.



Representação de uma Solução

Uma solução é representada por uma matriz $E_{|T| \times |D|}$, sendo:

- $|T| = 3$, são os três turnos possíveis: manhã, tarde, pernoite.
- $|D|$, são o número de dias do mês em que o escalonamento está sendo executado.



Abordagem Heurística

Representação de uma Solução

A Figura 1 exemplifica uma possível solução para o PGECTA em um mês hipotético de 4 dias.

$$s = \left[\begin{array}{c|cccc} & \text{Dia1} & \text{Dia2} & \text{Dia3} & \text{Dia4} \\ \hline \text{Manha} & e_{0,1} & e_{0,2} & e_{0,3} & e_{0,4} \\ \text{Tarde} & e_{1,1} & e_{1,2} & e_{1,3} & e_{1,4} \\ \text{Pernoite} & e_{2,1} & e_{2,2} & e_{2,3} & e_{2,4} \end{array} \right]$$

Figura: Exemplo de Solução para o PGECTA



Representação de uma Solução

Cada célula $e_{i,j}$ na matriz da Figura 1 é um vetor *booleano* de tamanho igual ao número de controladores n_{CTA} . Ou seja:

$$e_{i,j} = [cta_0, cta_1, cta_2, \dots, cta_{n_{CTA}}]$$

Desconsiderando-se as restrições operacionais, podemos exemplificar uma possível escala dos controladores para três turnos do mês hipotético exposto na solução da Figura 1:

$$e_{0,1} = [0, 1, 1, 0, 1]$$

$$e_{1,3} = [1, 1, 1, 1, 1]$$

$$e_{2,4} = [1, 0, 0, 0, 1]$$



Avaliação de uma Solução

No modelo heurístico adotado, uma solução s é avaliada por uma função f , a ser minimizada, dada pela Eq. (1). Esta função é baseada em penalidades, sendo composta de duas parcelas: f^{obj} e f^{inv} .

$$f(s) = f^{obj}(s) + f^{inv}(s) \quad (1)$$



Avaliação de uma Solução

A parcela f^{obj} , dada pela Eq. (2), refere-se à função objetivo do problema:

$$f^{obj}(s) = \sum_{c \in CTA} P_{c,t,d} \times x_{c,t,d} \quad \forall t \in T; d \in D \quad (2)$$

Como uma solução s pode não respeitar todas as restrições, penalizamos da seguinte forma:

$$f^{inv}(s) = \sum_{c \in CTA} \left(f_c^T(s) + f_c^{T_M, T, N}(s) + f_c^{PC}(s) + f_c^{SF}(s) + f_c^F(s) + f_c^{MD}(s) \right) \quad (3)$$



Estruturas de vizinhança

Para explorar o espaço de soluções do problema foram propostas duas estruturas de vizinhança:

Movimento Adiciona e Remove CTA - $N^{AR}(s)$: Este movimento consiste em dar folga a um dado CTA em um turno i do dia j e colocar um outro CTA para trabalhar no lugar deste. Desta maneira, neste movimento uma célula do vetor $e_{i,j}$ muda seu valor de 1 para 0, enquanto uma outra célula deste mesmo vetor mudar seu valor de 0 para 1.

$$e_{inicial(2,4)} = [1, 0, 0, 0, 1]$$
$$e_{final(2,4)} = [0, 0, 1, 0, 1]$$

Figura: Exemplo aplicação do movimento $N^{AR}(s)$



Estruturas de vizinhança

Movimento Troca CTA - $N^T(s)$: Consiste em trocar duas células distintas, ou seja, diferentemente do movimento $N^{AR}(s)$, neste movimento podemos selecionar um CTA de um vetor $e_{i,j}$ e outro CTA de um vetor $e_{n,m}$.

$$e_{inicial(1,2)} = [1, 0, 0, 1, 1]$$

$$e_{inicial(2,3)} = [1, 0, 1, 0, 1]$$

$$e_{final(1,2)} = [1, 0, 1, 0, 1]$$

$$e_{final(2,3)} = [1, 0, 0, 1, 1]$$

Figura: Exemplo aplicação do movimento $N^T(s)$



Algoritmo MSGVNS

Construção de uma Solução Inicial

Para geração de uma solução inicial para o problema utiliza-se uma heurística gulosa, cujo pseudocódigo está descrito no Algoritmo 2.

Algoritmo 2: ConstróiSoluçãoInicial()

Entrada: Conjunto de dias D a serem alocados

Saída: Solução de inicial S

```
1 enquanto  $D \neq \emptyset$  faça
2     Escolha um dia aleatório  $d \in D$ 
3     Remova o dia  $d$  do conjunto  $D$ 
4      $t \leftarrow$  manhã
5      $Rank_M \leftarrow$  GereRankMelhoresCTAs( $t, d$ )
6      $S(t, d) \leftarrow Rank_M$ 
7      $t \leftarrow$  tarde
8      $Rank_T \leftarrow$  GereRankMelhoresCTAs( $t, d$ )
9      $S(t, d) \leftarrow Rank_T$ 
10     $t \leftarrow$  noite
11     $Rank_N \leftarrow$  GereRankMelhoresCTAs( $t, d$ )
12     $S(t, d) \leftarrow Rank_N$ 
13 fim
14 Retorne  $S$ 
```



Algoritmo MSGVNS

Construção de uma Solução Inicial

Algoritmo 3: GereRankMelhoresCTAs

Entrada: Turno t , Dia d , Número de CTAs disponíveis $nCTA$

Saída: Vetor v dos melhores CTAs para trabalhar no turno t do dia d

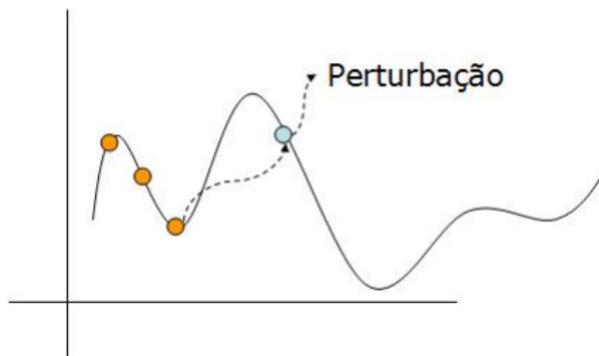
```
1 Rank[.]  $\leftarrow$  0
2 para  $c \leftarrow 1$  até  $nCTA$  faça
3     Rank[ $c$ ]  $\leftarrow$  foPenalizacao( $c, t, d$ )
4     Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foDescanso( $c, t, d$ )
5     Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foMaxDiasSemFolga( $c, t, d$ )
6     Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foMaxDiasFolga( $c, t, d$ )
7     Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foMaxTurnoFixoTrab( $c, t, d$ )
8     Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foMaxTurnosTrab( $c, t, d$ )
9     Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foMaxNoitesConsecutivas( $c, t, d$ )
10    Rank[ $c$ ]  $\leftarrow$  Rank[ $c$ ] + foMinTurnosTrab( $c, t, d$ )
11 fim
12  $v \leftarrow$  CTAs (necessários para atender o Turno  $t$ ) com menor valor referente ao vetor Rank
13 Retorne  $v$ 
```



Abordagem Heurística

Perturbação GVNS

Com a abordagem gulosa utilizada, têm-se um número de $D!$ possibilidades de alocar os controladores.
Em um dado nível p , aplica-se $p + 1$ movimentos aleatórios.



Algoritmo MSGVNS

Pseudocódigo Básico

Algoritmo 4: MSGVNS

Entrada: *MultiStartmax*, *IterMax*, Função $f(\cdot)$

Saída: Solução s^* de qualidade possivelmente superior à s de acordo com a função f

```

1   $s_0 \leftarrow$  Melhor Solução em MultiStartmax iterações do procedimento ConstróiSoluçãoInicial()
2   $s^* \leftarrow$  VND( $s_0$ ,  $f$ )
3   $p \leftarrow 0$ 
4  enquanto critério de parada não satisfeito faça
5      iter  $\leftarrow 0$ 
6      enquanto iter < IterMax e critério de parada não satisfeito faça
7           $s' \leftarrow$  Refinamento( $s^*$ ,  $p$ ,  $f$ )
8          se  $s'$  for melhor que  $s^*$  de acordo com a função  $f$  então
9               $s^* \leftarrow s'$ ;
10              $p \leftarrow 0$ ;
11             iter  $\leftarrow 0$ 
12             fim
13             senão
14                 iter  $\leftarrow$  iter + 1
15             fim
16         fim
17          $p \leftarrow p + 1$ 
18 fim
19 retorna  $s^*$ 
    
```



Abordagem Heurística

Refinamento

Algoritmo 5: Refinamento

Entrada: r vizinhanças: N^{AR} , N^T

Entrada: Solução Inicial s , Nível p e Função de Avaliação f

Saída: Solução s

- 1 **para** $i \leftarrow 1$ até $p + 1$ **faça**
 - 2 $k \leftarrow$ SeleccioneVizinhança(r)
 - 3 $s' \leftarrow$ Perturbação(s , k)
 - 4 **fim**
 - 5 $s \leftarrow$ VND(s' , f)
 - 6 **retorna** s
-



Resultados

- Algoritmos implementados em C++
- PC Pentium Core 2 Quad (Q9550) com 2,83 GHz e 8GB de RAM
- Compilador g++ 4.2.4
- Foram utilizados dois conjuntos de instâncias.
 - 1 r55_1 e r55_2, associadas à operação de um centro de controladores de tráfego aéreo brasileiro, considerando dados disponibilizados na Internet
 - 2 v53_1 e v51_1, geradas neste trabalho



Resultados

Framework OptFrame

O algoritmo proposto foi implementando com auxílio do *framework* OptFrame¹. Essa escolha, foi devido a seu arcabouço de fácil utilização, que inclui:

- Representações de soluções e populações;
- Arcabouços de métodos heurísticos e metaheurísticos;
- Interface para análise dos resultados;
- Fórum de discussão com os colaboradores do projeto.

¹disponível em <http://sourceforge.net/projects/optframe/>



Resultados

Característica das Instâncias

Tabela: Característica das Instâncias

Instância	<i>nCTA</i>	<i>nDias</i>	<i>nRestricoesCTA</i>
r55_1	55	30	43,85
r55_2	55	30	45,72
v53_1	53	30	48
v51_1	51	30	50



Resultados

Pesos e Parâmetros

Após uma bateria de testes preliminares, foram utilizados os seguintes parâmetros:

- Fase de construção da solução inicial:
 - 1 $foDescanso = 5000$
 - 2 $foMaxDiasSemFolga = 5000$
 - 3 $foMaxTurnoFixoTrab = 5000$
 - 4 $foMaxTurnosTrab = 5000$
 - 5 $foMaxNoitesConsecutivas = 5000$
 - 6 $foMaxDiasFolga = -200$
 - 7 $foMinTurnosTrab = -200$
- Função objetivo:
 - 1 inviabilidades: $f_c^T(s)$, $f_c^{T,M,T,N}(s)$, $f_c^{PC}(s)$, $f_c^{SF}(s)$, $f_c^F(s)$ e $f_c^{MD}(s) = 10000$
- parâmetros dos métodos *Multi-Start* e GVNS:
 - 1 $MultiStartmax = 300000$ e $IterMax = 50$.



Resultados

Foram realizadas 100 execuções para cada uma das instâncias com tempo limite de 15s.

Tabela: Resultados *MSGVNS*

Instância	Tempo (s)	MSGVNS	
		Melhor	Média
r55_1	15	0	4,47
r55_2	15	0	10,85
v53_1	15	176	195,99
v51_1	15	578	801,13



Resultados

Características das melhores soluções

Tabela: Características das melhores soluções

Instância	$nMaxTurnos$	$nMinTurnos$	$nRestricoesVioladas$
r55_1	19	18	0
r55_2	19	18	0
v53_1	20	19	4
v51_1	20	20	18



Conclusões

- Utilizando dois problemas-teste envolvendo dados reais e dois gerados neste trabalho, o algoritmo proposto, MSGVNS, foi capaz de encontrar soluções de boa qualidade rapidamente e com baixa variabilidade das soluções finais
- Em dois dos problemas-teste, o algoritmo proposto conseguiu provar a otimalidade de duas soluções



Trabalhos Futuros

- Criação de um conjunto mais amplo de problemas-testes
- Concessão de novos tipos de perturbações e ampliação da busca local
- Implementação de um modelo matemático para o problema, buscando:
 - 1 uma boa comparação entre o algoritmo proposto *MSGVNS* e um *solver* comercial
 - 2 implementação de um algoritmo híbrido, combinando a flexibilidade das metaheurísticas com o poderio da programação matemática.



Produtos gerados

- 1 MSGVNS: UM ALGORITMO HEURÍSTICO PARA O PROBLEMA DE GERENCIAMENTO DE ESCALA OPERACIONAL DE CONTROLADORES DE TRÁFEGO AÉREO. In: *X Simpósio de Pesquisa em Transporte Aéreo*, 2011, Ouro Preto. 2011, p.1-10.



Agradecimentos

- UFOP
- Prof. Túlio Ângelo Machado Toffolo
- Prof. Marcone Jamilson Freitas Souza



Obrigado pela atenção!



Perguntas?

