



UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"



RESOLUÇÃO DO PROBLEMA DE ESCALONAMENTO DE ENFERMEIROS EM HOSPITAL UTILIZANDO A METAHEURÍSTICA SIMULATED ANNEALING

Aluno: Lucas Henrique Russo do Nascimento

Orientadora: Prof. Dra. Andréa Carla Gonçalves Vianna

SUMÁRIO



1 - INTRODUÇÃO

2 - PROBLEMA

3 - REVISÃO BIBLIOGRÁFICA

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA

5 - MÉTODOS DE RESOLUÇÃO

6 - GERADOR DE INSTÂNCIAS

7 - TESTES COMPUTACIONAIS

8 - CONCLUSÕES

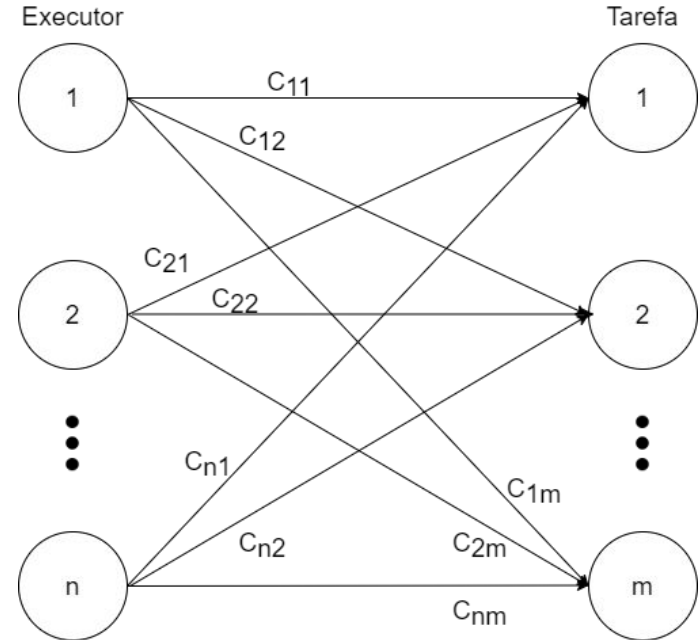
REFERÊNCIAS

1 - INTRODUÇÃO

1 - INTRODUÇÃO

- Problema de Escalonamento (ou Alocação):
 - atribuir executores à tarefas
 - cada execução da tarefa possui um custo c_{ij}
- Objetivo: executar todas as tarefas com o menor custo possível
- Motivações:
 - agilidade na entrega de algum serviço
 - atender prazos
 - aumentar lucro
 - melhorar algum serviço oferecido

Imagem 1 - representação do problema de alocação



Fonte: elaborado pelo autor

1 - INTRODUÇÃO

- Problema de Alocação de Navios em Berços (SILVA, 2008)



- Aplicação da metaheurística Busca Tabu no Problema de Alocação de Aulas a Salas em uma universidade (SUBRAMANIAN *et al.*, 2011)
- Problema de escalonamento de médicos no Hospital das Clínicas de Porto Alegre (MEISTER, 2020)

Médico	Dia 1			Dia 2		
	Manhã	Tarde	Noite	Manhã	Tarde	Noite
M1	A1	-	-	A1	-	-
M2	A1	-	-	-	A1	-
M3	-	A2	-	A2	-	-
M4	-	-	A2	-	-	A2
M5	A2	-	-	A2	-	-
M6	A2	-	-	A1	-	-
M7	-	A1	-	A1	-	-
M8	-	-	A1	-	-	A1
Para cada dia e turno é indicado se o médico terá folga (-) ou em qual área será alocado						

Fonte: Meister, 2020

1 - INTRODUÇÃO



Este trabalho aborda:

- Problema de Escalonamento de Enfermeiros
- **Modelagem** matemática do problema
- **Validação** da modelagem através de um *solver*
- Proposta de uma **metaheurística** para resolver o problema

—

2 - PROBLEMA

2 - PROBLEMA



Problema de Escalonamento de Enfermeiros se resume em montar a escala de trabalho dos enfermeiros, a saber:

- Horário de funcionamento ininterrupto do hospital (24h)
- Quantidade mínima e máxima de enfermeiros e técnicos por período
- Possíveis restrições:
 - carga horária do funcionário
 - dias de descanso
 - preferências por turno
 - recursos do hospital
- Busca-se atingir a maior eficiência do atendimento médico, minimizando custos do hospital



3 - REVISÃO BIBLIOGRÁFICA

3 - REVISÃO BIBLIOGRÁFICA



Problema de Escalonamento de Pessoal (PEP)

- “processo de planejamento ou organização de horários de trabalho de funcionários, considerando regulamentos laborais, organizacionais, contratuais ou sindicais” (Devesse, 2016)
 - “o processo de construção de horários de trabalho para funcionários, a fim de satisfazer a demanda de bens ou serviços de uma organização” (Ernst *et al.*, 2004)
 - Modelagem da **demand**a
 - Definição das **folgas**
 - Definição dos **turnos**
 - Definição das **tarefas**
 - Definição do **pessoal**
- Nem todos os problemas podem precisar de todos esses módulos.

3 - REVISÃO BIBLIOGRÁFICA



Problema de Escalonamento de Médicos (PEM) X Problema de Escalonamento de Enfermeiros (PEE)

(Devesse, 2016)

- Ambos derivam do PEP
- **Contratos** diferentes, **regras** diferentes
- **Vínculo** ou não com a instituição médica (ressalva para hospital escola)
- Maximizar **preferências**/diminuir **custos**
- **Restrições** diferentes (ex: área de atuação do médico)

3 - REVISÃO BIBLIOGRÁFICA



Problema de Escalonamento de Enfermeiros (PEE) (Cheang *et al.*, 2003)

- Padrões de modelagem:
 - *nurse-day view*
 - *nurse-task view*
 - *nurse-shift pattern*
- Tipos de restrições
 - Factibilidade
 - Qualidade

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA



Definição:

Determinar a **escala** de trabalho de enfermeiros em um determinado **período**, a partir de uma **instância**, considerando algumas **restrições**, acordos contratuais, legislação trabalhista, regras da instituição e preferências pessoais de cada profissional.

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA



Instância:

Conjunto de informações que definem algumas restrições

- Período a ser analisado
- Quantidade mínima e máxima de cada dia/turno
- Lista de enfermeiros do hospital, cada um contendo:
 - Nome
 - Carga horária do período

No problema, são considerados **3 turnos diários**, sendo manhã, tarde e noite, todos com duração de **8 horas**.

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA



Restrições:

- **R1 - Demanda mínima:** quantidade de enfermeiros mínima por turno de cada dia.
- **R2 - Demanda máxima:** quantidade de enfermeiros máxima por turno de cada dia.
- **R3 - Turnos consecutivos proibidos:** um enfermeiro alocado no turno da noite não pode ser alocado no turno da manhã ou da tarde do dia seguinte.
- **R4 - Carga horária:** a soma do total de horas alocadas para um médico em todo o período deve ser igual a sua carga horária
- **R5 - Máximo de noites consecutivas:** quando o enfermeiro possui mais de 3 noites alocadas em sequência, a solução é penalizada.
- **R6 - Apenas um turno por dia:** cada enfermeiro só pode trabalhar um turno em cada dia.

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA



Restrições:

- Podem ter pesos diferentes na penalização da solução
- CPLEX: Restrições de **factibilidade**
- Metaheurística: Restrições de **qualidade**

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA

Modelagem:

Quadro 1: Variáveis de entrada

Símbolo	Definição
$m \in M$	médico/enfermeiro m pertencente ao conjunto M de médicos/enfermeiros do hospital
$d \in D$	dia d pertencente ao conjunto D de dias a serem alocados
$t \in T$	turno t pertencente ao conjunto T de turnos a serem alocados
$dMin_{dt}$	demanda mínima de médicos no dia d no turno t
$dMax_{dt}$	demanda máxima de médicos no dia d no turno t
qH_t	quantidade de horas do turno t
ch_m	carga horária do médico/enfermeiro m

Fonte: elaborado pelo autor.

Quadro 2: Variáveis de decisão

Símbolo	Definição
$x_{mdt} \in \{0, 1\}$	1 se o médico/enfermeiro m está alocado no dia d no turno t 0 caso contrário

Fonte: elaborado pelo autor.

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA

Modelagem:

$$\min \sum_{m \in M} \sum_{d \in D} \sum_{t \in T} C_{mdt} * x_{mdt} \quad (3.1)$$

Sujeito à:

R1 - Demanda mínima

$$\sum_{m \in M} x_{mdt} \geq dMin_{dt} \quad \forall d \in D, t \in T \quad (3.2)$$

R2 - Demanda máxima

$$\sum_{m \in M} x_{mdt} \leq dMax_{dt} \quad \forall d \in D, t \in T \quad (3.3)$$

4 - DEFINIÇÃO E MODELAGEM DO PROBLEMA

Modelagem:

R3 - Turnos subsequentes

$$x_{md3} + \sum_{t=1}^2 x_{m(d+1)t} \leq 1 \quad \forall m \in M, d \in \{1, \dots, |D| - 1\} \quad (3.4)$$

R4 - Carga horária

$$\sum_{m \in M} x_{mdt} * qH_t \leq ch_m \quad \forall d \in D, t \in T \quad (3.5)$$

R5 - Três noites consecutivas

$$x_{md3} + x_{m(d+1)3} + x_{m(d+2)3} + x_{m(d+3)3} \leq 3 \quad \forall m \in M, d \in \{1, \dots, |D| - 3\} \quad (3.6)$$

R6 - Apenas um turno por dia

$$\sum_{t \in T} x_{mdt} \leq 1 \quad \forall m \in M, d \in D \quad (3.7)$$

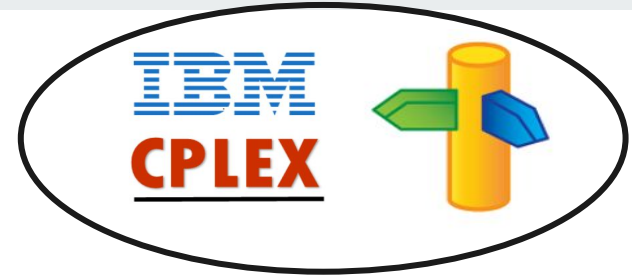
$$x_{mdt} \in \{0, 1\} \quad (3.8)$$

5 - MÉTODOS DE RESOLUÇÃO

5 - MÉTODOS DE RESOLUÇÃO



Solver IBM CPLEX:



Solver é um software de **programação matemática** que resolve **problemas de otimização**.

- Arquivos .dat e .mod

Linguagem OPL (Optimization Programming Language):

- declarações de inicialização de dados
- pré-processamento
- restrições
- função objetivo
- pós-processamento

O solver foi utilizado apenas para testar e **validar a modelagem**.

5 - MÉTODOS DE RESOLUÇÃO

Simulated Annealing:

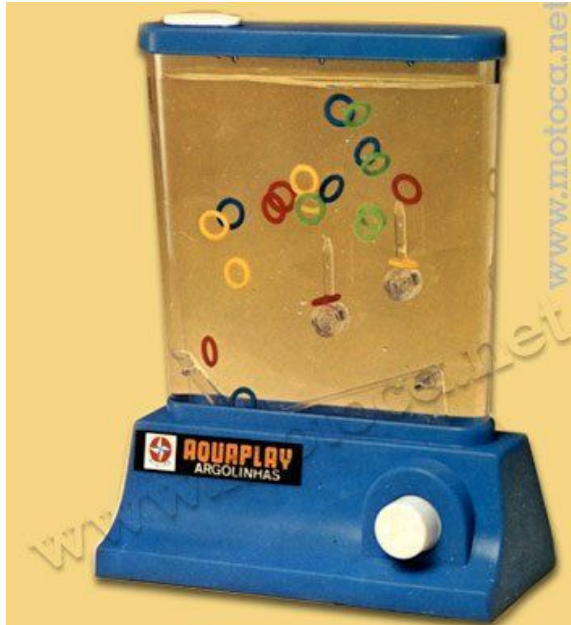
- Recozimento ou arrefecimento: processo metalúrgico para resfriamento de metais
 - aquecer o metal em altas temperaturas: moléculas ficam agitadas
 - resfriamento lento e gradual: as moléculas conseguem se acomodar de forma organizada, deixando o material menos quebradiço. (Prata, 2018)



5 - MÉTODOS DE RESOLUÇÃO

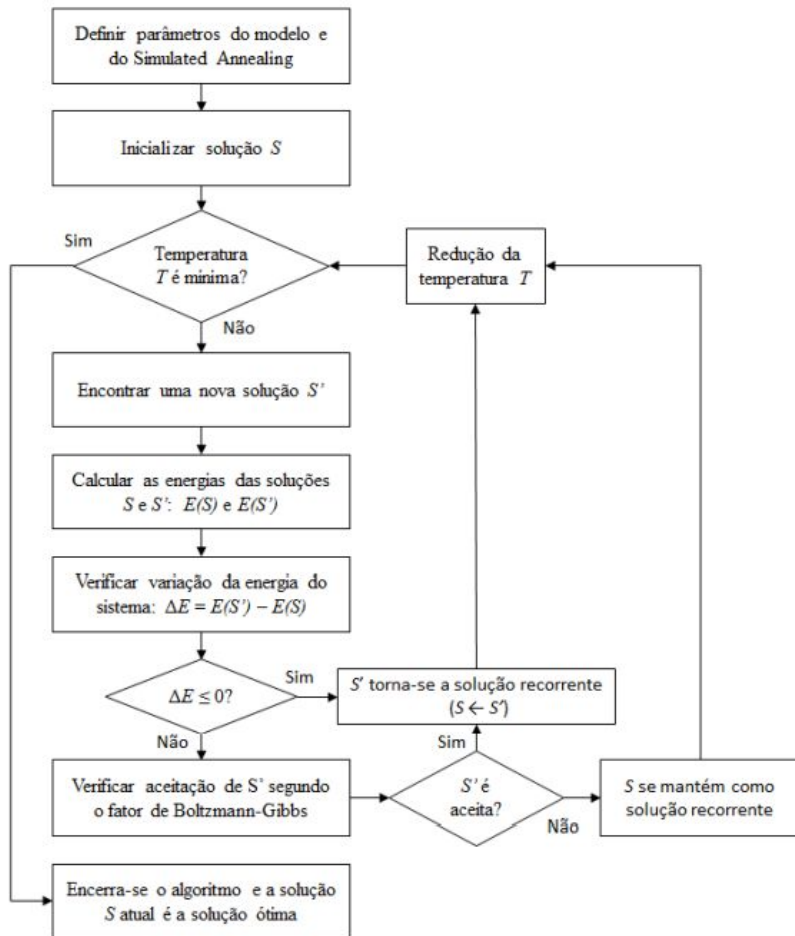
Simulated Annealing:

- Analogia com brinquedo (Adhikari, 2017)



- Movimentos iniciais aleatórios
- Aceita movimentos de piora da solução no início (evita casos de ótimos locais)
- Conforme as argolas vão encaixando, aumenta o cuidado para não bagunçar o que já encaixou

Simulated Annealing:



- solução inicial aleatória
- temperatura inicial: 100.000
- temperatura final: 0,000001
- taxa de resfriamento: 1%

- Fator de Boltzmann-Gibbs:

$$\beta = \min\{1, e^{-\Delta E / T}\}$$

Limit

$$\lim_{x \rightarrow \infty} e^{-x/y} = 0 \text{ for } y > 0$$

$$\lim_{y \rightarrow \pm \infty} e^{-x/y} = 1$$

6 - GERADOR DE INSTÂNCIAS

6 - GERADOR DE INSTÂNCIAS



Facilitar e automatizar os testes.

Gera um arquivo no formato .JSON

O arquivo é composto basicamente de três propriedades:

- as informações relativas ao período a ser analisado (**Period**)
- uma lista com as informações relativas aos enfermeiros (**Physicians**)
- uma lista relativa a informação dos turnos (**Shifts**)

```
{
  "Period":{
    "Year":2010,
    "Month":9,
    "StartDay":1,
    "EndDay":30
  },
  "Physician":[
    {
      "Id":0,
      "Name":"Ana",
      "Hours":240
    }
  ],
  "Shifts":[
    {
      "Id":0,
      "Hours":8,
      "DMin":1,
      "DMax":10
    }
  ]
}
```

7 - TESTES COMPUTACIONAIS

7 - TESTES COMPUTACIONAIS

CPLEX:

- 7 enfermeiros (A - G), carga horária de 40 horas
- 5 dias
- 3 turnos de 8 horas
- demanda mínima: 2
- demanda máxima: 5

R1 - Demanda mínima

R2 - Demanda máxima

R3 - Turnos consecutivos proibidos

R4 - Carga horária

R5 - Máximo de noites consecutivas

R6 - Apenas 1 turno por dia

Quadro 3: Resultado da alocação com instância reduzida

	Dia 1	Dia 2	Dia 3	Dia 4	Dia 5
Turno 1	D, E	D, E	D, E	C, D	B, D
Turno 2	A, F	C, F	C, F	B, F	C, G
Turno 3	B, G	A, G	A, G	A, E	E, F

Fonte: Elaborado pelo autor.

7 - TESTES COMPUTACIONAIS



Metaheurística:

4 testes feitos:

1. solução inicial aleatória e melhorada
2. porcentagem de variação no espaço de busca
3. variação da penalização
4. testes gerais com instâncias variadas

7 - TESTES COMPUTACIONAIS

Metaheurística:

1. Solução inicial aleatória e melhorada

- teste com instância **reduzida**

Tabela 1: Comparação de execução com diferentes soluções iniciais para instância reduzida

Execução	Solução inicial aleatória		Solução inicial melhorada	
	Quantidade de violações na solução inicial	Quantidade de violações na solução final	Quantidade de violações na solução inicial	Quantidade de violações na solução final
1	40	1	6	1
2	46	0	6	0
3	39	1	6	1
4	37	1	6	0
5	38	2	6	1

Fonte: Elaborado pelo autor.

7 - TESTES COMPUTACIONAIS

Metaheurística:

1. Solução inicial aleatória e melhorada

- teste com instância **aumentada**

Tabela 2: Comparação de execução com diferentes soluções iniciais para instância aumentada

Execução	Solução inicial aleatória		Solução inicial melhorada	
	Quantidade de violações na solução inicial	Quantidade de violações na solução final	Quantidade de violações na solução inicial	Quantidade de violações na solução final
1	369	2	42	3
2	325	2	42	0
3	330	0	42	3
4	346	3	42	0
5	342	4	42	1

Fonte: Elaborado pelo autor.

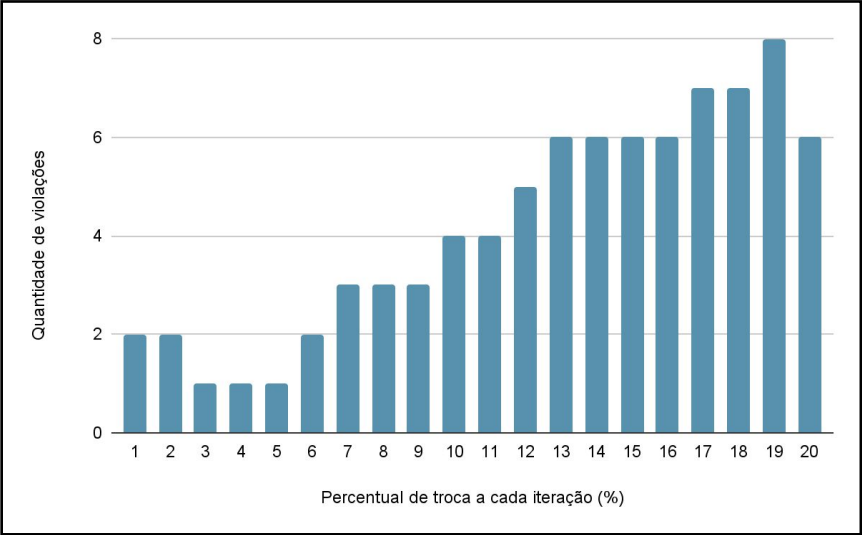
7 - TESTES COMPUTACIONAIS

Metaheurística:

2. Variação no espaço de busca

- teste com instância **reduzida**

Gráfico 1: Variação da quantidade de violações da solução final para instância reduzida



Fonte: elaborado pelo autor.

Tabela 3: Variação percentual no espaço de busca para instância reduzida

Percentual de troca (%)	Valor correspondente à quantidade de alocações trocadas	Violações na solução inicial	Violações na solução final
1	1	30	2
2	2	37	2
3	3	34	1
4	4	46	1
5	5	46	1
6	6	38	2
7	7	55	3
8	8	45	3
9	9	45	3
10	10	37	4
11	12	40	4
12	13	51	5
13	14	38	6
14	15	44	6
15	16	49	6
16	17	53	6
17	18	45	7
18	19	43	7
19	20	39	8
20	21	30	6

Fonte: Elaborado pelo autor.

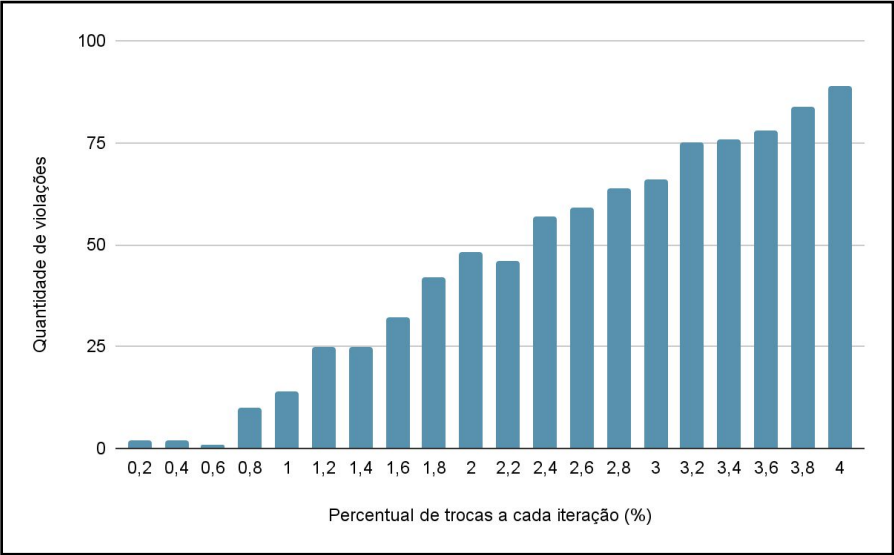
7 - TESTES COMPUTACIONAIS

Metaheurística:

2. Variação no espaço de busca

- teste com instância **aumentada**

Gráfico 2: Variação da quantidade de violações da solução final para instância aumentada



Fonte: elaborado pelo autor.

Tabela 4: Variação percentual no espaço de busca para instância aumentada

Percentual de troca (%)	Valor correspondente à quantidade de alocações trocadas	Violações na solução inicial	Violações na solução final
0.2	2	349	2
0.4	4	327	2
0.6	5	340	1
0.8	7	338	10
1	9	363	14
1.2	11	316	25
1.4	13	359	25
1.6	14	325	32
1.8	16	355	42
2	18	416	48
2.2	20	304	46
2.4	22	351	57
2.6	23	365	59
2.8	25	300	64
3	27	354	66
3.2	29	336	75
3.4	31	351	76
3.6	32	361	78
3.8	34	359	84
4	36	342	89

Fonte: elaborado pelo autor.

7 - TESTES COMPUTACIONAIS

Metaheurística:

3. Variação da penalização

- teste com R1 e R2

R1 - Demanda mínima

R2 - Demanda máxima

R3 - Turnos consecutivos proibidos

R4 - Carga horária

R5 - Máximo de noites consecutivas

R6 - Apenas 1 turno por dia

Tabela 5: Quantidade de violações por restrição nas soluções inicial e final, R1 e R2 com peso maior

Exe- cução	Quantidade de violações por restrição na solução inicial					Quantidade de violações por restrição na solução final				
	R1 R2	R3	R4	R5	R6	R1 R2	R3	R4	R5	R6
1	16	156	10	31	162	0	19	0	0	15
2	18	138	10	14	146	0	16	0	1	20
3	17	142	10	17	154	0	20	0	0	21
4	14	148	10	26	154	0	20	0	0	17
5	17	149	10	9	147	0	20	0	1	18
6	22	136	10	11	143	0	15	0	0	21
7	19	146	10	21	153	0	19	0	1	12
8	13	162	10	15	161	0	11	0	0	16
9	20	147	10	17	159	0	16	0	1	18
10	20	151	10	23	150	0	20	0	0	15

Fonte: Elaborado pelo autor.

7 - TESTES COMPUTACIONAIS

Metaheurística:

3. Variação da penalização



- teste com R3

R1 - Demanda mínima
R2 - Demanda máxima
R3 - Turnos consecutivos proibidos
R4 - Carga horária
R5 - Máximo de noites consecutivas
R6 - Apenas 1 turno por dia

Tabela 6: Quantidade de violações por restrição nas soluções inicial e final, R3 com peso maior

Exe- cução	Quantidade de violações por restrição na solução inicial					Quantidade de violações por restrição na solução final				
	R1 R2	R3	R4	R5	R6	R1 R2	R3	R4	R5	R6
1	23	160	10	23	154	13	0	0	0	14
2	19	147	10	14	154	19	0	0	0	7
3	19	141	10	11	136	12	0	0	0	9
4	25	156	10	30	158	10	0	0	1	22
5	15	144	10	16	149	20	1	0	0	11
6	14	142	10	24	146	16	0	0	0	14
7	15	135	10	12	146	16	0	0	0	10
8	21	149	10	21	148	10	0	0	2	14
9	17	138	10	14	147	13	0	0	1	14
10	10	145	10	13	154	11	0	0	1	13

Fonte: Elaborado pelo autor.

7 - TESTES COMPUTACIONAIS

Metaheurística:

3. Variação da penalização

- teste com R6

R1 - Demanda mínima
R2 - Demanda máxima
R3 - Turnos consecutivos proibidos
R4 - Carga horária
R5 - Máximo de noites consecutivas
R6 - Apenas 1 turno por dia

Tabela 7: Quantidade de violações por restrição nas soluções inicial e final, R6 com peso maior

Exe- cução	Quantidade de violações por restrição na solução inicial					Quantidade de violações por restrição na solução final				
	R1 R2	R3	R4	R5	R6	R1 R2	R3	R4	R5	R6
1	18	160	10	29	149	14	13	0	0	0
2	20	135	10	15	147	11	17	0	0	0
3	15	142	10	18	164	12	11	0	1	0
4	20	131	10	19	148	8	13	0	0	0
5	22	151	10	22	156	13	16	0	0	0
6	14	148	10	10	156	12	16	0	0	0
7	9	135	10	5	137	13	11	0	0	0
8	22	1443	10	24	140	12	24	0	0	1
9	13	146	10	26	142	15	11	0	0	0
10	13	129	9	13	143	15	13	0	0	0

Fonte: Elaborado pelo autor.

7 - TESTES COMPUTACIONAIS

Metaheurística:

4. Testes gerais

Instâncias geradas:

Quadro 4: Demandas mínima e máxima separadas por instância

Instâncias	Quantidade de enfermeiros	Demanda mínima			Demanda máxima		
		Turno 1	Turno 2	Turno 3	Turno 1	Turno 2	Turno 3
Inst_1	11	3	2	3	8	10	8
Inst_2	11	2	2	3	10	10	10
Inst_3	12	3	3	3	7	10	6
Inst_4	12	2	3	3	6	9	6
Inst_5	9	2	3	3	8	8	10
Inst_6	11	2	3	3	10	10	9
Inst_7	11	2	2	3	8	6	10
Inst_8	11	2	3	3	9	10	6
Inst_9	8	3	3	2	10	9	7
Inst_10	8	3	2	2	9	10	8

Fonte: elaborado pelo autor.

7 - TESTES COMPUTACIONAIS

Metaheurística:

4. Testes gerais



Tabela 8: Quantidade de violações por instância, separadas por restrição

Instância	Quantidade de violações por restrição na solução inicial					Quantidade de violações por restrição na solução final				
	R1 R2	R3	R4	R5	R6	R1 R2	R3	R4	R5	R6
Inst_1	5	177	11	25	191	11	0	0	0	0
Inst_2	2	161	11	16	166	7	1	0	0	1
Inst_3	16	164	12	17	181	16	0	0	0	0
Inst_4	17	169	12	20	174	9	1	0	0	1
Inst_5	8	130	9	13	133	25	0	0	0	1
Inst_6	0	156	11	14	156	11	1	0	0	2
Inst_7	14	172	11	32	174	7	1	0	0	1
Inst_8	13	157	11	30	165	12	0	0	0	0
Inst_9	13	114	8	9	118	34	0	0	0	1
Inst_10	3	114	8	9	116	20	0	0	0	0

Fonte: Elaborado pelo autor.

R1 - Demanda mínima

R2 - Demanda máxima

R3 - Turnos consecutivos proibidos

R4 - Carga horária

R5 - Máximo de noites consecutivas

R6 - Apenas 1 turno por dia

7 - TESTES COMPUTACIONAIS

Metaheurística:

4. Testes gerais

- **R1 e R2** com peso maior

R1 - Demanda mínima
R2 - Demanda máxima
R3 - Turnos consecutivos proibidos
R4 - Carga horária
R5 - Máximo de noites consecutivas
R6 - Apenas 1 turno por dia

Tabela 9: Quantidade de violações por instância, separadas por restrição, R1 e R2 com peso maior

Instâncias	Quantidade de violações por restrição na solução inicial					Quantidade de violações por restrição na solução final				
	R1 R2	R3	R4	R5	R6	R1 R2	R3	R4	R5	R6
Inst_1	1	160	11	25	171	0	5	0	0	6
Inst_2	1	162	11	23	155	0	1	0	1	1
Inst_3	12	152	12	15	162	0	14	0	0	1
Inst_4	32	191	12	34	200	0	8	0	0	7
Inst_5	6	125	9	15	122	0	24	2	0	20
Inst_6	2	162	11	19	168	0	9	0	0	7
Inst_7	16	176	11	17	179	0	3	0	1	6
Inst_8	10	160	11	20	158	0	15	0	0	1
Inst_9	13	105	8	10	111	1	35	5	0	40
Inst_10	10	111	8	16	114	0	30	2	0	25

Fonte: Elaborado pelo autor.

8 - CONCLUSÕES

8 - CONCLUSÕES



- Solução inicial \longrightarrow aleatória
 - possível motivo: início do algoritmo tem muitos movimentos de piora da solução
- Espaço de busca \longrightarrow pouca variação
 - permite refinar melhor a solução
- Pesos diferentes para violações diferentes é eficaz

De maneira geral, a metaheurística *Simulated Annealing* mostrou-se **eficaz** para a resolução do problema.

8 - CONCLUSÕES



Trabalhos futuros:

- Estudo de caso com instâncias reais, mais restrições.
- Proposição de metaheurística híbrida (ex: *Simulated Annealing* com *Variable Neighborhood Search*)

REFERÊNCIAS

REFERÊNCIAS



ADHIKARI, B. **The simulated annealing algorithm explained with an analogy to a toy**. 2017. 1 video (11 min). Disponível em: <https://www.youtube.com/watch?v=eBmU1ONJ-os>. Acesso em: 8 nov 2021.

BESSEGATO, L.; PEREIRA, A.; TOLEDO, L. 2016. Planejamento econômico de controle on-line por atributo com intervalo de inspeção variável e classificações repetidas. **EMEPRO**. Disponível em: https://www.researchgate.net/publication/309357358_Planejamento_economico_de_controle_on-line_por_atributo_com_intervalo_de_inspecao_variavel_e_classificacoes_repetidas . Acesso em: 08 nov 2021.

CHEANG, B.; LI, H.; LIM, A.; RODRIGUES, B. Nurse rostering problems - a bibliographic survey. **European Journal of Operational Research**, v. 151, n.2, p. 447-460, 2003.

DEVESSE, V. A. P. A. **Métodos de solução para o problema de escalonamento de médicos**. Orientadora: Maristela Oliveira dos Santos. 2016. Dissertação (Mestrado em Ciências da Computação) - Universidade de São Paulo. Disponível em: https://teses.usp.br/teses/disponiveis/55/55134/tde-07112016-101859/publico/ValdemarAbraoPedroAnastacioDevesse_revisada.pdf . Acesso em: 1 ago 2022.

ERNST, A.; JIANG, H.; KRISHNAMOORTHY, M.; SIER, D. Staff scheduling and rostering: A review of applications, methods and models. **European Journal of Operational Research**, v.153,n.2,p.3-27,2004.

REFERÊNCIAS



MEISTER, T. C. **Um algoritmo heurístico aplicado ao problema de escalonamento de médicos do Hospital de Clínicas de Porto Alegre**. Orientadora: Luciana Buriol. 2020. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Rio Grande do Sul, Rio Grande do Sul. 2020. Disponível em: <https://www.lume.ufrgs.br/bitstream/handle/10183/217592/001121923.pdf?sequence=1>. Acesso em: 06 mai. 2021.

PRATA, B. **Aula 6 - Meta-heurísticas (Simulated Annealing)**. 2018. 1 video (17 min). Disponível em: <https://www.youtube.com/watch?v=4VGt0jN73fc>. Acesso em: 8 nov 2021.

SILVA, V. M. D. **Um modelo heurístico para alocação de navios em berços**. Orientador: Antônio Sérgio Coelho, 2008. Dissertação (Mestrado em Engenharia de Produção) - Universidade Federal de Santa Catarina, Santa Catarina. 2008. Disponível em: <https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/91781/249010.pdf?sequence=1&isAllowed=y>. Acesso em: 06 mai. 2021.

SUBRAMANIAN, Anand et al. Aplicação da metaheurística busca tabu ao problema de alocação de aulas a salas em uma instituição universitária. **Revista Produção Online**, Florianópolis, v. 11, n. 1, p. 54-75, mar. 2011. ISSN 16761901. Disponível em: <https://www.producaoonline.org.br/rpo/article/view/419/762>. Acesso em: 06 mai 2021.

—

OBRIGADO!