UNIVERSIDADE FEDERAL DE MINAS GERAIS
Escola de Engenharia
Curso de Bacharelado em Engenharia de Sistemas

Raphael Henrique Braga Leivas 2020028101
Milton Pereira Bravo Neto 2018072549
Daniel Felipe de Almeida Araújo 2023422617

Trabalho Computacional 01 – Teoria da Decisão (ELE088)

Belo Horizonte
2025

SUMÁRIO

1 INTRODUÇÃO 3
2 MODELAGEM 4
2.1 Problema 1 4
2.2 Problema 2 4
2.3 Modelagem Multiobjetivo 5
3 ALGORITMOS 6
4 RESULTADOS E DISCUSSÃO 8
5 CONCLUSÃO 9
6 REFERÊNCIAS 10

1 INTRODUÇÃO

2 MODELAGEM

Temos que modelar dois problemas mono-objetivos:

- Problema 1: minimização do custo de manutenção total $f_1(\cdot)$
- Problema 2: minimização do custo esperado de falha total $f_2(\cdot)$

2.1 Problema 1

Temos essencialmente um problema de designação simples. Seja N o número de equipamentos e J o número de políticas de manutenção, definimos a variável de decisão x_{ij} por

$$x_{ij}: \text{ se a máquina } i \text{ executa a manutenção } j \tag{2.1}$$

onde

$$x_{ij} \in \{0,1\} \quad , \quad i = \{1,2,...,N\} \quad , \quad j = \{1,2,...,J\}$$

Para a função objetivo, seja c_j o custo de executar a manutenção j. Note que esse custo independe da máquina i que estamos executando a manutenção. Temos a função objetivo

$$\min f_1 = \sum_{i=1}^{N} \sum_{j=1}^{J} c_j x_{ij} \tag{2.2}$$

sujeito a

$$\sum_{j=1}^{J} x_{ij} = 1 \quad , \quad \forall i = 1,2,...,N \tag{2.3}$$

A Equação 2.3 indica que toda máquina executa exatamente uma política de manutenção. Além disso, note que solução de Equação 2.3 é trivial: basta escolher o plano de manutenção com o menor custo para todos os equipamentos.

2.2 Problema 2

O custo da falha de cada equipamento é dado pelo produto da probabilidade de falha p_{ij} pelo custo da falha do equipamento, dada por d_i. Assim, temos

$$\min f_2 = \sum_{i=1}^{N} \sum_{j=1}^{J} p_{ij}\, d_i\, x_{ij} \tag{2.4}$$

onde

$$x_{ij} \in \{0,1\} \quad , \quad i = \{1,2,...,N\} \quad , \quad j = \{1,2,...,J\}$$

$$p_{ij} = \frac{F_i\left(t_0 + k_j \Delta t\right) - F_i\left(t_0\right)}{1 - F_i\left(t_0\right)} \quad , \quad F_i(t) = 1 - \exp\left[-\left(\frac{t}{\eta_i}\right)^{\beta_i}\right]$$

sujeito a

$$\sum_{j=1}^{J} x_{ij} = 1 \quad , \quad \forall i = 1,2,...,N \tag{2.5}$$

Note que na Equação 2.4 temos essencialmente um problema de programação linear inteira.

2.3 Modelagem Multiobjetivo

Juntando as modelagens dos problemas mono-objetivos acima, temos o a modelagem multiobjetivo do problema.

$$\min f_1 = \sum_{i=1}^{N} \sum_{j=1}^{J} c_j x_{ij}$$

$$\min f_2 = \sum_{i=1}^{N} \sum_{j=1}^{J} p_{ij}\, d_i\, x_{ij}$$

x_{ij} : se a máquina i executa a manutenção j

sujeito a

$$\sum_{j=1}^{J} x_{ij} = 1 \quad , \quad \forall i = 1,2,...,N$$

$$x_{ij} \in \{0,1\} \quad , \quad i = \{1,2,...,N\} \quad , \quad j = \{1,2,...,J\}$$

3 ALGORITMOS

O Algoritmo 1 mostra a versão do BNVS implementada no trabalho

Algoritmo 1 BVNS implementado no trabalho.

```
procedure BVNS(x, k_max)
    while num_sol_avaliadas < max_sol_avaliadas do
        k ← 1
        while k < k_max do
            x' ← SHAKE(x, k)
            x'' ← FIRSTIMPROVEMENT(x, x', k)
            x, k ← NEIGHBORHOODCHANGE(x, x'', k)
        end while
    end while
end procedure
```

O Algoritmo 2 mostra a função Shake.

Algoritmo 2 Função Shake.

```
                    ▷ Gera uma solução aleatória na k-ésima estrutura de vizinhança.
procedure SHAKE(x, k)
    if k = 1 then
        y ← 1-swap
    end if
    if k = 2 then
        y ← Permutaçao de dois planos de manutenção
    end if
    if k = 3 then
        y ← Mudança de um bloco de máquinas para outro plano
    end if

    return y
end procedure
```

O Algoritmo 3 mostra a heurística construtiva.

Algoritmo 3 Heurística construtiva para gerar a solução inicial.

```
procedure SOLUCAOINICIAL()
    x ← Solução aleatória
    for all i in x do
        if variancia(p_ij d_i) ≥ limite then
            x[i] ← Manutenção mais cara
        else
            x[i] ← Manutenção mais barata
        end if
    end for

    return x
end procedure
```

4 RESULTADOS E DISCUSSÃO

5 CONCLUSÃO

6 REFERÊNCIAS