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

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

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

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 do equipamento 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 todo equipamento 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 o equipamento 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. Nela estão definidas as três estruturas de vizinhança escolhidas para implementação. As duas primeiras são vizinhaças de refinamento, mas com abordagens diferentes. E a terceira é uma vizinhança de perturbação para buscar sair de mínimos locais:

- A primeira estrutura é o que chamamos de um movimento de *1-swap*, onde é escolhido aleatorimente um equipamento e trocado seu plano para um dos outros dois restantes, a escolha do plano também é aleatória.
- A segunda estrutura é a troca ou permutação dos planos de dois equipamentos diferentes escolhidos também aleatóriamente.
- A terceira estrutura por sua vez, altera um bloco de 50 equipamentos em sequência, onde o início do bloco é aleatório. Nesse bloco é avaliado qual o plano mais comum e troca-se o plano de manutenção de todos os integrantes do bloco para um mesmo plano, diferente do mais comum encontrado anteriormente.

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 ← Permutação de dois planos de manutenção
    end if
    if k = 3 then
        y ← Mudança de um bloco de equipamentos para outro plano
    end if

    return y
end procedure
```

O Algoritmo 3 mostra a heurística construtiva utilizada para a criação da solução inicial. Basicamente o probema se reduz em escolher um plano de manutenção, dentre os três disponíveis, para cada equipamento minimizando o custo da manutenção e o custo de falha dos equipamentos. A minimização do custo da manutenção se dá escolhendo a manutenção mais barata para todos os equipamentos, e a minimização do custo de falha escolhendo a manutenção mais cara. Olhando para o segundo problema, temos a matriz de custos de falha $p_{ij}d_i$ onde i é cada equipamento e j os planos de manutencão, para cada equipamento i fixo avalia-se a variância de $p_{ij}d_i$ e caso esse valor seja maior que o limiar de 0.5 escolhe a manutenção mais cara para compor a solução inicial daquele equipamento, caso seja menor que o limiar é escolhida a manutenção mais barata. A lógica envolvida é que se o custo de falha não varia tanto para aquele equipamento, não é necessário a manutenção mais cara.

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) ≥ limiar 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