



**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Tarefa DWDM

RTR029007 - REDES DE TRANSMISSÃO (2025 .1 - T01)

Prof. Fábio Alexandre de Souza

Aluno: Wagner Flores dos Santos

Julho de 2025

Sumário

1. Objetivo	3
2. Fundamento Teórico	3
3. Descrição da Solução	3
4. Detalhes da Implementação	3
5. Código comentado	3
6. Exemplo de Execução	5
7. Conclusão	7

1. Objetivo

O objetivo deste projeto é implementar uma aplicação em Python que permita analisar o fenômeno de Four Wave Mixing (FWM) em sistemas de transmissão óptica baseados na grade ITU-T G.692 (Dense Wavelength Division Multiplexing - DWDM). A aplicação simula a geração de produtos de intermodulação entre sinais e detecta possíveis sobreposições interferentes.

2. Fundamento Teórico

O fenômeno de Four Wave Mixing ocorre quando três ondas ópticas interagem no meio não-linear da fibra e geram uma quarta frequência definida por:

$$f_f = f_i + f_j - f_k, \text{ com } i \neq k \text{ e } j \neq k$$

Caso a frequência gerada f_f coincida com uma frequência utilizada no sistema, ocorre interferência por sobreposição, prejudicando a transmissão dos dados.

3. Descrição da Solução

A solução foi desenvolvida em Python e oferece uma interface por menu (estilo “do Wille”), onde o usuário pode:

1. Visualizar os canais disponíveis na grade ITU G.692 (100 GHz).
2. Selecionar os canais a serem utilizados na transmissão e executar a análise automática de FWM, verificando combinações interferentes.

4. Detalhes da Implementação

A grade de canais segue a recomendação ITU-T G.692, com canais de 17 a 61 e espaçamento de 0.1 THz e as frequências são representadas em THz.

Todas as combinações de três frequências distintas são testadas para gerar um possível produto de FWM e o resultado é arredondado para uma casa decimal para evitar problemas de ponto flutuante.

O programa verifica se o valor gerado f_f coincide com qualquer frequência de entrada.

5. Código comentado

```
from itertools import permutations # Importa função para gerar permutações
de elementos

# Grade padrão DWDM (ITU-T G.692) com espaçamento de 100 GHz
```

```

# Cria um dicionário onde a chave é o número do canal e o valor é a frequência
em THz
base_canal = 191.7 # Frequência do canal 17
canais_itu = {canal: round(base_canal + 0.1 * (canal - 17), 1) for canal in
range(17, 62)}

def listar_canais_itu():
    """Exibe a lista de canais disponíveis da grade ITU-T G.692."""
    print("\n--- Canais ITU-T G.692 ---")
    for canal, freq in canais_itu.items():
        print(f"Canal {canal}: {freq} THz")
    print("-" * 40)

def escolher_e_analisar():
    """Permite ao usuário escolher canais e realiza a análise de FWM."""

    # Recebe os números dos canais digitados pelo usuário
    entrada = input("Digite os números dos canais desejados (ex: 20 21 23): ")

    try:
        # Converte os números digitados para inteiros
        canais = list(map(int, entrada.strip().split()))

        # Verifica se todos os canais estão dentro da grade válida
        for c in canais:
            if c not in canais_itu:
                raise ValueError(f"Canal {c} não está na grade ITU-T G.692.")

        # Verifica se há ao menos 3 canais (mínimo para gerar FWM)
        if len(canais) < 3:
            print("Erro: é necessário escolher pelo menos 3 canais.\n")
            return

        # Converte os canais em frequências correspondentes (em THz)
        frequencias = [canais_itu[c] for c in canais]
        frequencias_set = set(frequencias) # Conjunto para busca rápida

        interferencias = set() # Guarda os casos de interferência detectados

        # Gera todas as permutações possíveis de 3 frequências diferentes
        for fi, fj, fk in permutations(frequencias, 3):
            ff = round(fi + fj - fk, 1) # Calcula frequência gerada
            e arredonda

            # Verifica se a frequência gerada ff está entre as frequências
            de entrada

            if ff in frequencias_set:
                interferencias.add((fi, fj, fk, ff)) # Adiciona a
            interferência

        # Exibe o resultado da análise
        print("\n--- ANÁLISE FWM ---")

```

```

        if interferencias:
            print(f"Detectadas {len(interferencias)} sobreposições por Four
Wave Mixing:\n")
            for fi, fj, fk, ff in sorted(interferencias):
                print(f"ff = {fi} + {fj} - {fk} = {ff} THz (interferente!)")
        else:
            print("Nenhuma interferência FWM detectada.")
        print("-" * 40)

    except Exception as e:
        # Captura e exibe erros de entrada inválida
        print(f"Erro: {e}")

def menu():
    """Exibe o menu principal e gerencia as opções do usuário."""
    while True:
        print("\n=== MENU FWM - ITU G.692 ===")
        print("1. Listar canais ITU disponíveis")
        print("2. Escolher canais e analisar Four Wave Mixing")
        print("3. Sair")
        opcao = input("Escolha uma opção: ")

        if opcao == '1':
            listar_canais_itu() # Mostra canais disponíveis
        elif opcao == '2':
            escolher_e_analisar() # Escolhe e analisa canais
        elif opcao == '3':
            print("Encerrando o programa.")
            break # Sai do laço e encerra
        else:
            print("Opção inválida. Tente novamente.")

# Inicia o programa com o menu interativo
menu()

```

6. Exemplo de Execução

```

=== MENU FWM - ITU G.692 ===
1. Listar canais ITU disponíveis
2. Escolher canais e analisar Four Wave Mixing
3. Sair
Escolha uma opção: 1

--- Canais ITU-T G.692 ---
Canal 17: 191.7 THz
Canal 18: 191.8 THz
Canal 19: 191.9 THz
Canal 20: 192.0 THz
Canal 21: 192.1 THz
Canal 22: 192.2 THz

```

Canal 23: 192.3 THz
Canal 24: 192.4 THz
Canal 25: 192.5 THz
Canal 26: 192.6 THz
Canal 27: 192.7 THz
Canal 28: 192.8 THz
Canal 29: 192.9 THz
Canal 30: 193.0 THz
Canal 31: 193.1 THz
Canal 32: 193.2 THz
Canal 33: 193.3 THz
Canal 34: 193.4 THz
Canal 35: 193.5 THz
Canal 36: 193.6 THz
Canal 37: 193.7 THz
Canal 38: 193.8 THz
Canal 39: 193.9 THz
Canal 40: 194.0 THz
Canal 41: 194.1 THz
Canal 42: 194.2 THz
Canal 43: 194.3 THz
Canal 44: 194.4 THz
Canal 45: 194.5 THz
Canal 46: 194.6 THz
Canal 47: 194.7 THz
Canal 48: 194.8 THz
Canal 49: 194.9 THz
Canal 50: 195.0 THz
Canal 51: 195.1 THz
Canal 52: 195.2 THz
Canal 53: 195.3 THz
Canal 54: 195.4 THz
Canal 55: 195.5 THz
Canal 56: 195.6 THz
Canal 57: 195.7 THz
Canal 58: 195.8 THz
Canal 59: 195.9 THz
Canal 60: 196.0 THz
Canal 61: 196.1 THz

=== MENU FWM - ITU G.692 ===

1. Listar canais ITU disponíveis
2. Escolher canais e analisar Four Wave Mixing
3. Sair

Escolha uma opção: 2

Digite os números dos canais desejados (ex: 20 21 23): 20 21 22

--- ANÁLISE FWM ---

Detectadas 2 sobreposições por Four Wave Mixing:

$ff = 192.0 + 192.2 - 192.1 = 192.1 \text{ THz}$ (interferente!)

$ff = 192.2 + 192.0 - 192.1 = 192.1 \text{ THz}$ (interferente!)

7. Conclusão

A aplicação permite, de forma interativa e didática, a identificação de frequências geradas por FWM que podem causar sobreposição e interferência em sistemas DWDM. A estrutura modular do código facilita sua ampliação, podendo ser integrada futuramente a sistemas reais ou simulações mais completas com análise gráfica e exportação de relatórios.