

Laboratório 09: MMI

Professor Adolfo Fernandes Herbster

Aluno

Matrícula

1 Objetivos

- gerar os componentes e objetos de simulação por meio da API Python;
- realizar análises do dispositivo por meio do API Python;
- utilizar o Lumerical EME para determinar o comprimento ótimo da caixa central de um MMI;
- comparar o desempenho do MMI simulado no EME com os resultados obtidos por meio do `varFDTD` e FDTD;
- utilizar o Lumerical INTERCONNECT para simular o dispositivo utilizando os parâmetros S gerados no EME e FDTD.

2 Atividades

- Considere o dispositivo 2×2 MMI (*Multi-mode Interferometer*) em Si ilustrado na Fig. 1, utilizado como *splitter* 50/50, com largura da caixa central $W_M = 7 \mu\text{m}$ e comprimento dos *tapers* igual a $15 \mu\text{m}$ e largura maior igual a $1,5 \mu\text{m}$. Todos os guias monomodos possuem dimensões de $450 \times 220 \text{ nm}$ (largura x altura). Lembre-se de realizar todas as tarefas deste guia por meio da API Python.

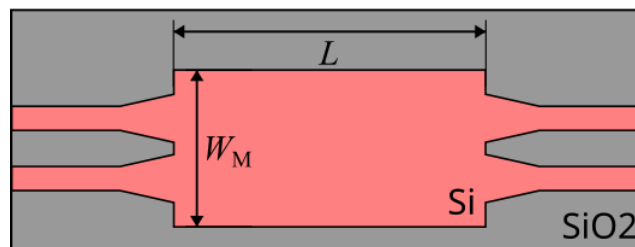


Figura 1: MMI em Si.

2.1 Modelo de simulação

1. **Cálculo do comprimento da caixa central** - A partir da equação:

$$L = \frac{3L\pi}{N} \quad (1)$$

determine o comprimento da caixa central do MMI.

2. **Criação do dispositivo via API** - Crie o dispositivo, ilustrado na Fig. 1, no Lumerical MODE por meio da API Python.
3. **Adição do solver, fontes e monitores** - Adicione o solver do tipo EME. Não esqueça de definir o parâmetro **background material** como **SiO2 (Glass) - Palik**. Lembre-se que o parâmetro **subcell method** deve ser do tipo **CVCS** apenas para os *tapers*.

Para melhorar a precisão dos resultados, inclua um objetivo tipo **Mesh** na região dos *tapers*. Lembre-se de realizar estas tarefas por meio da API Python.

1 Atividades

2.1 Modelo de simulação

. Utilizando a fórmula apresentada no guia foi determinado que o comprimento L da caixa central para a formação de duas imagens $N = 2$ é:

$$L = \frac{3 \cdot L_{\pi}}{2} = 229.41111684744303 \mu m$$

A criação do dispositivo via API foi feita conforme mostrado no trecho de código 1

```
1 #box
2 emeApi.addrect()
3 emeApi.addtogroup(dev_layer)
4 emeApi.set('name', 'box')
5 emeApi.set('x', 0)
6 emeApi.set('x span', L)
7 emeApi.set('y', 0)
8 emeApi.set('y span', Wm)
9 emeApi.set('z', height/2)
10 emeApi.set('z span', height)
11 emeApi.set('material', materialSi)
12 #tapers
13 for i in range(4):
14     emeApi.addobject('linear_taper')
15     emeApi.set('name', f'taper port {str(i+1)}')
16     emeApi.set('thickness', height)
17     emeApi.set('angle_side', 90)
18     emeApi.set('width_l', width)
19     emeApi.set('width_r', taper_large_tip)
20     emeApi.set('len', taper_length)
21     emeApi.set('material', materialSi)
22     emeApi.set('x', -(emeApi.getnamed(f'{dev_layer}::box', 'x max') +
23         taper_length/2))
24     emeApi.set('y', Wm/4)
25     emeApi.set('z', height/2)
26 #adjusting port 2
27 emeApi.select(f'{dev_layer}::taper port 2')
28 emeApi.set('width_l', taper_large_tip)
29 emeApi.set('width_r', width)
30 emeApi.set('x', emeApi.getnamed(f'{dev_layer}::box', 'x max') +
31     taper_length/2)
32 emeApi.set('y', Wm/4)
33 ##adjusting port 3
34 emeApi.select(f'{dev_layer}::taper port 3')
35 emeApi.set('width_l', taper_large_tip)
36 emeApi.set('width_r', width)
37 emeApi.set('x', emeApi.getnamed(f'{dev_layer}::box', 'x max') +
38     taper_length/2)
39 emeApi.set('y', -Wm/4)
40 #adjusting port 4
41 emeApi.select(f'{dev_layer}::taper port 4')
42 emeApi.set('y', -Wm/4)
43 #waveguides
44 for i in range(4):
```

```

45     emeApi.addrect()
46     emeApi.addtogroup(dev_layer)
47     emeApi.set('name', f'port {str(i+1)}')
48     emeApi.set('x', -(emeApi.getnamed(f'{dev_layer}::box', 'x max') +
taper_length + length/2))
49     emeApi.set('x span', length)
50     emeApi.set('y', Wm/4)
51     emeApi.set('y span', width)
52     emeApi.set('z', height/2)
53     emeApi.set('z span', height)
54     emeApi.set('material', materialSi)
55 #adjusting
56 emeApi.select(f'{dev_layer}::port 2')
57 emeApi.set('x', (emeApi.getnamed(f'{dev_layer}::box', 'x max') +
taper_length + length/2))
58
59 emeApi.select(f'{dev_layer}::port 3')
60 emeApi.set('x', (emeApi.getnamed(f'{dev_layer}::box', 'x max') +
taper_length + length/2))
61 emeApi.set('y', -Wm/4)
62
63 emeApi.select(f'{dev_layer}::port 4')
64 emeApi.set('y', -Wm/4)

```

2.2 Simulação - Solver EME

. A distribuição do campo ao utilizar o solver EME é exibido na Figura 1 e na Figura 2, temos o perfil de campo em dB.

Figura 1: Perfil de campo - EME

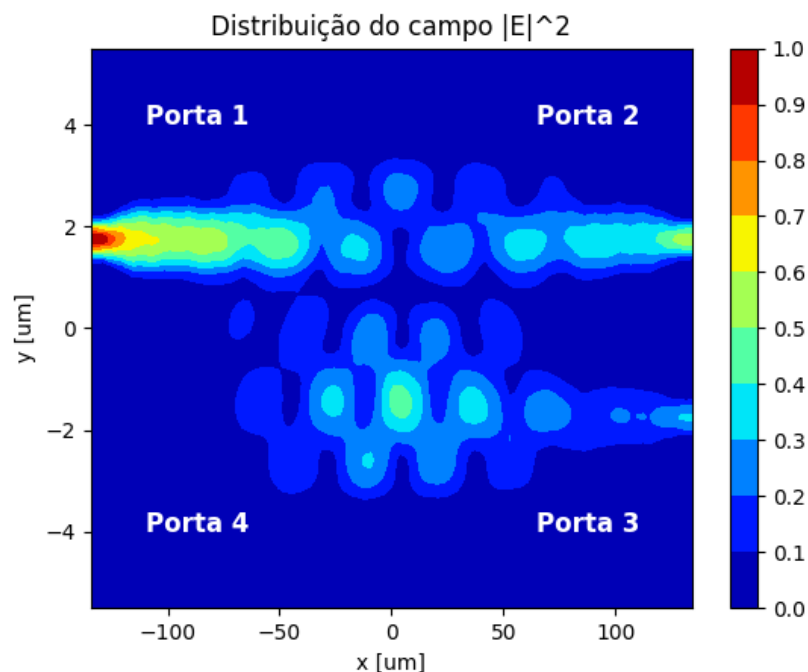
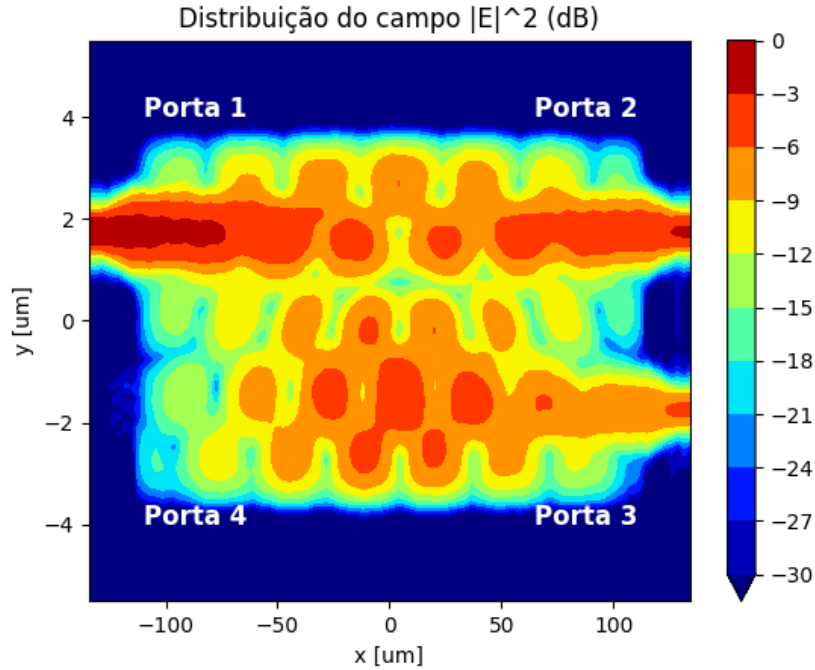


Figura 2: Perfil de campo - EME (dB)



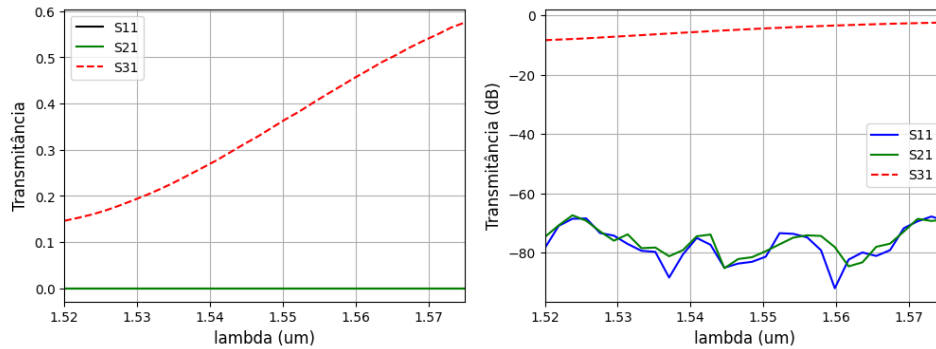
2.2.1 Simulação do dispositivo

Percebe-se que os comprimentos do guia central para a formação de N imagens se aproxima bastante dos valores calculador a partir da Equação 1, para 2 e 3 imagens temos, respectivamente, $229.4\mu m$ e $152.94\mu m$.

2.2.2 Desempenho em função do comprimento de onda

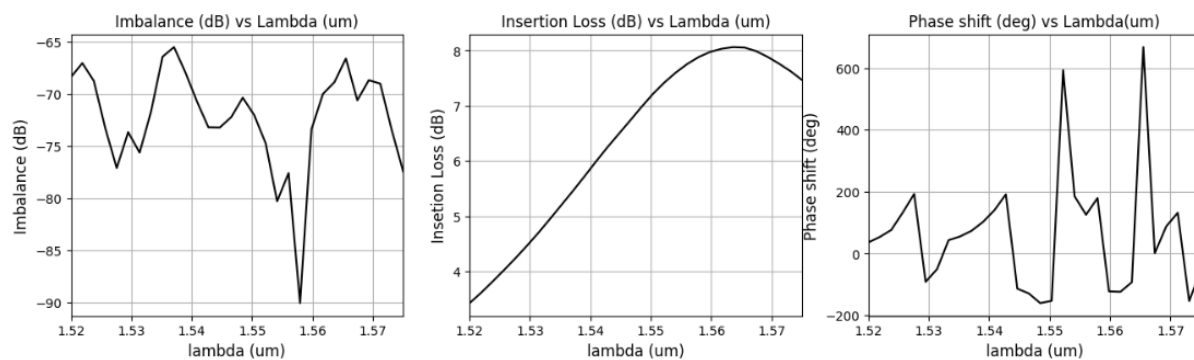
Na Figura 3, temos o gráfico que representa o desempenho do dispositivo (Transmitância) em função do comprimento de onda na escala linear e logarítmica.

Figura 3: Transmitância ($229.411\mu m$)



Na Figura 4, temos perda de inserção, desbalanceamento e o erro de fase do dispositivo.

Figura 4: Perdas ($229.411\mu m$)



2.2.3 Otimização do comprimento da caixa central

Após a conclusão do item 2.2.2 foi realizado um ajuste no comprimento do guia central ($L = 235\mu m$), e os resultados são mostrados na Figura 5 (Transmitância) e na Figura 6 são exibidos os gráficos relacionados as perdas.

Figura 5: Transmitância ($235\mu m$)

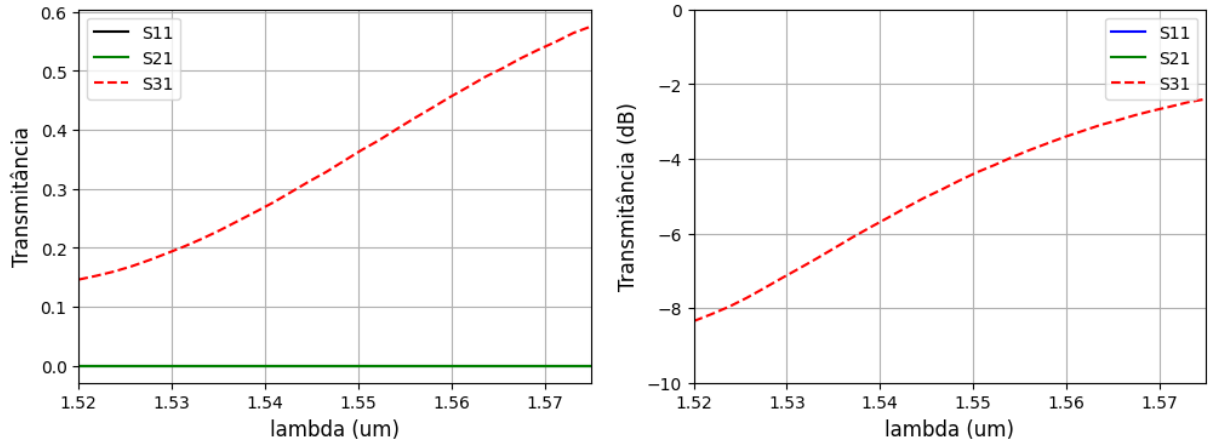
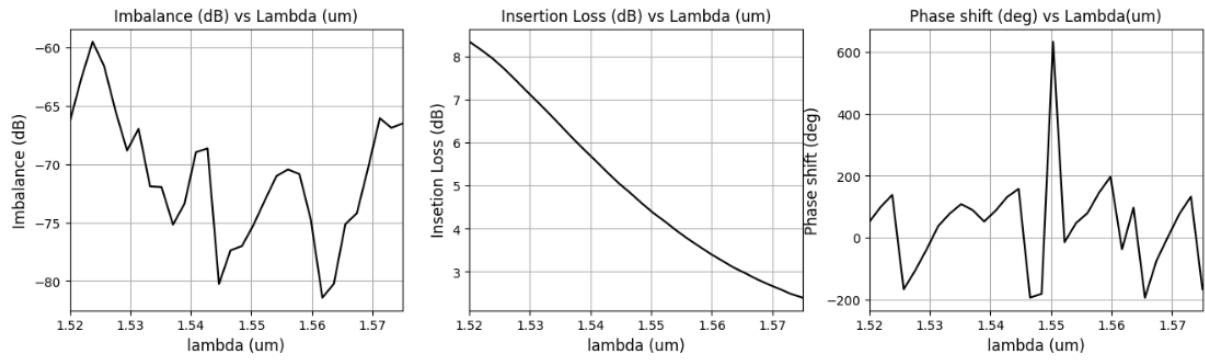


Figura 6: Perdas ($235\mu m$)



2.3 Simulação - solver varFDTD

2.3.1 Criação do dispositivo e objetos via script - varFDTD

Foi utilizado o mesmo dispositivo criado na seção anterior. Além da alteração das fronteiras conforme a transmissão do modo desejado também foram feitas alterações de customização das fronteiras PML para garantir que a simulação evitasse divergência e ao mesmo tempo diminuir a reflexão. Tais alterações são mostradas no trecho de código 1.

```
1 #more stability
2 varApi.set('dt stability factor', 0.9)
3 #custom PML setup
4 varApi.set('PML type', 'stretched coordinate PML')
5 varApi.set('PML profile', 4)
6 varApi.set('PML layers', 40)
7 varApi.set('PML kappa', 10)
8 varApi.set('PML sigma', 1)
9 varApi.set('PML alpha', 10)
```

2.3.2 Simulação do dispositivo - varFDTD

A distribuição do campo ao utilizar o solver varFDTD é exibido na Figura 7 e na Figura 8, temos o perfil de campo em dB.

Figura 7: Perfil de campo - varFDTD

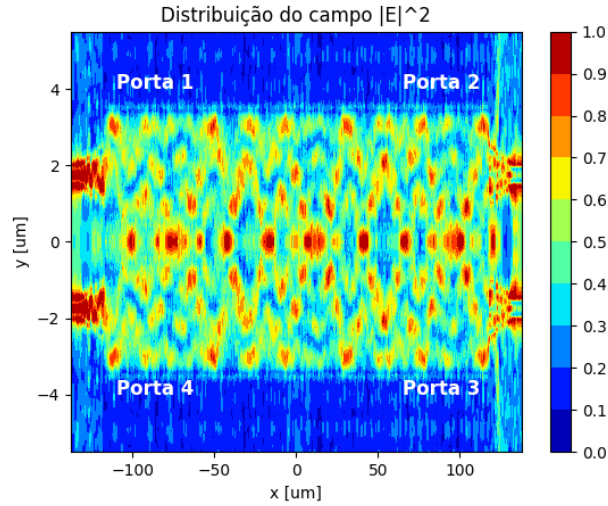
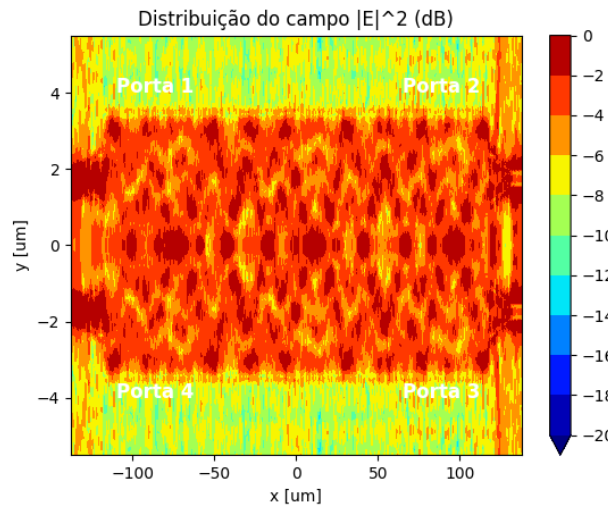
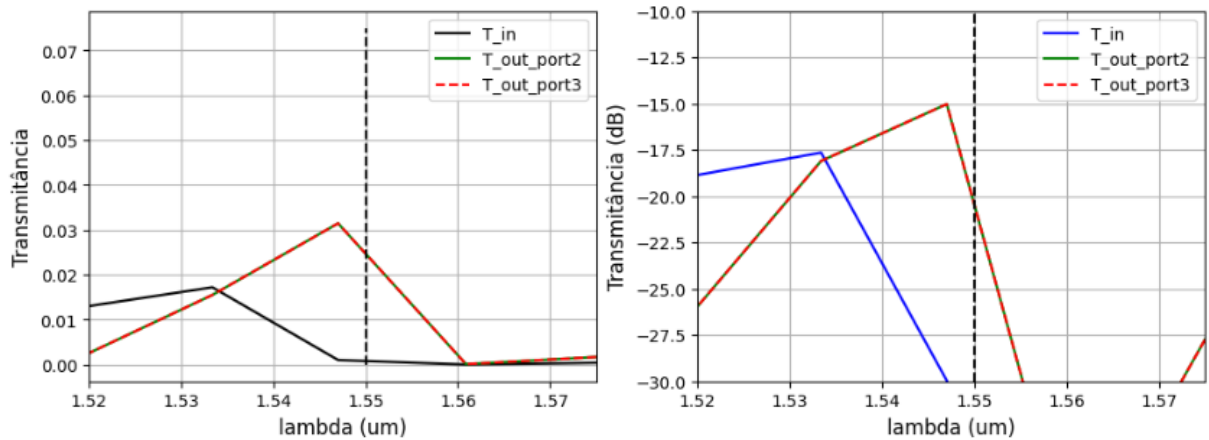


Figura 8: Perfil de campo - varFDTD (dB)



Na Figura 9 temos os gráficos de transmissão na escala linear e logarítmica.

Figura 9: Transmitância



Analisando os gráficos 7, 8, 9 percebe-se que o dispositivo não desempenha de forma ideal, mesmo com as otimizações de parâmetros recomendadas pela desenvolvedora do software de simulação.

2.3.3 Simulação do dispositivo - modo TM -varFDTD

Nas Figuras 10 e 11 são apresentados os perfis de campo em escala linear e escala logarítmica.

Figura 10: Perfil de campo do modo TM

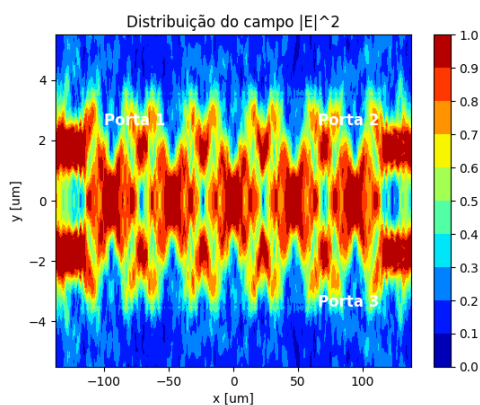
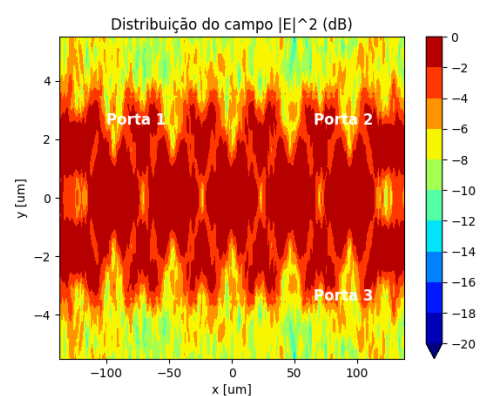


Figura 11: Perfil de campo do modo TM dB



2.4 Simulação - solver FDTD

2.4.1 Criação do dispositivo e objetos via script - FDTD

O dispositivo e objetos utilizados são os mesmos dos itens anteriores com a mesma ressalva do solver varFDTD em que foi necessário realizar adaptações das configurações de contorno a fim de evitar uma divergência da simulação.

2.4.2 Simulação do dispositivo - FDTD

Nas Figuras 12 e 13 são apresentados os perfis de campo em escala linear e escala logarítmica para o solver FDTD.

Figura 12: Perfil de campo do modo TM

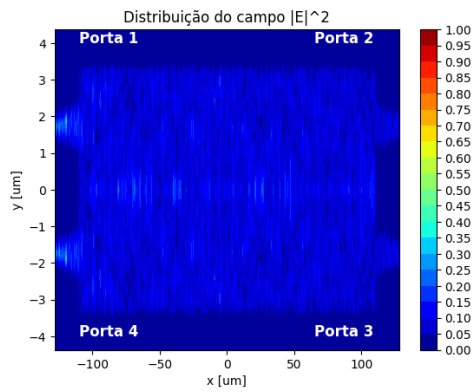
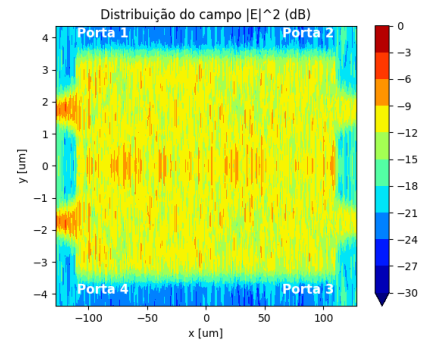
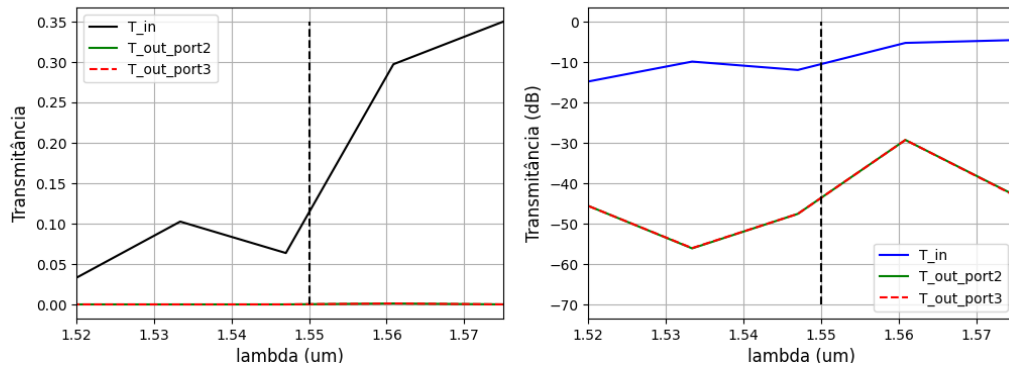


Figura 13: Perfil de campo do modo TM dB



A transmitância é apresentada na Figura ?? tanto na escala logarítmica quanto na linear.

Figura 14: Transmitância - FDTD (modo TE)



Observações

Ao analisar os perfis de campo, os gráficos relacionados a transmissão e perdas, principalmente dos solvers varFDTD e FDTD, fica claro que os parâmetros que foram alterados não foram o suficiente para garantir um dispositivo que desempenhasse o objetivo traçado (splitter 50/50), resultando em um dispositivo com baixos valores de transmissão e muita reflexão.