



UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE ENGENHARIA ELÉTRICA

TEEE: PROJETO DE CIRCUITOS FOTÔNICOS EM SILÍCIO

Atividade 7

Professor: Adolfo Fernandes Herbster
Aluno: Erick Cândido Sousa
Turma: 01
Matrícula: 120110361

Campina Grande - CG
2024

1. Objetivos.

- gerar o componentes e objetos de simulação por meio do *script*;
- utilizar o *Lumerical varFDTD* para simular o dispositivo (*y branch*);
- utilizar o *Lumerical FDTD* para: – obter os desempenho final do dispositivo; – comparar os resultados gerados com aqueles gerados pelo *varFDTD* e; – gerar seus parâmetros S;
- utilizar o *Lumerical INTERCONNECT* para simular o dispositivo utilizando os parâmetros S gerados no FDTD.

2. Atividades.

2.1. Criação de estrutura, *solver* e monitores.

- A primeira atividade consiste em gerar um dispositivo *y branch* de *Si* a partir de quatro guias curvados (*bend*) de raio $5\ \mu\text{m}$. Além disso, em cada porta do *y branch* é adicionado guias retos de $450 \times 220\ \text{nm}$. É ilustrado na Figura 1 a estrutura gerada.

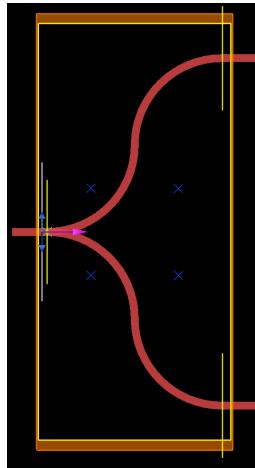


Figura 1 – *y branch* em *Si*.

- Para a criação do dispositivo, o seguinte código foi feito:

```
deleteall();
switchtolayout();
wg_width = 450e-9;
wg_height = 220e-9;
wg_radius = 5e-6;
#criar os bends
addobject("90_bend_wg");
set("name", "upper1");
set("first axis", 'z');
set("base width", wg_width);
set("base height", wg_height);
set("radius", wg_radius);
set("rotation 1", 90);
set("x", 0);
set("y", 0);
set("z", 0);
addobject("90_bend_wg");
set("name", "upper2");
set("first axis", 'z');
set("base width", wg_width);
```

```

set("base height", wg_height);
set("radius", wg_radius);
set("rotation 1", -90);
set("x", -10e-6);
set("y", 0);
set("z", 0);
addobject("90_bend_wg");
set("name", "upper3");
set("first axis", 'z');
set("base width", wg_width);
set("base height", wg_height);
set("radius", wg_radius);
set("rotation 1", 180);
set("x", 0);
set("y", -10e-6);
set("z", 0);
addobject("90_bend_wg");
set("name", "upper4");
set("first axis", 'z');
set("base width", wg_width);
set("base height", wg_height);
set("radius", wg_radius);
set("rotation 1", 0);
set("x", -10e-6);
set("y", -10e-6);
set("z", 0);
#criar os guias retangulares
addrect();
set("name", "out 1");
set("x span", 2e-6);
set("y span", wg_width);
set("z span", wg_height);
set("material", "Si (Silicon) - Palik");
set("x", 1e-6);
set("y", 5e-6);
set("z", 0);
addrect();
set("name", "out2");
set("x span", 2e-6);
set("y span", wg_width);
set("z span", wg_height);
set("material", "Si (Silicon) - Palik");
set("x", 1e-6);
set("y", -15e-6);
set("z", 0);
addrect();
set("name", "in");
set("x span", 2e-6);
set("y span", wg_width);
set("z span", wg_height);
set("material", "Si (Silicon) - Palik");
set("x", -11e-6);
set("y", -5e-6);
set("z", 0);

```

- Note que foram criadas 7 estruturas: 4 *bends* e 3 guias retangulares. Os *bends* possuem largura igual a 450 nm e altura igual a 220 nm. Para criar os *bends* foi utilizada a função “`addobject(“90_bend_wg”);`” e para criar os guias retangulares foi utilizada a função “`addrect();`”. As formas de dimensionamento são idênticas. Esse código foi salvo com o nome “ATIVIDADE_CREATE”. Essa informação será importante, visto que outros códigos irão chamar essa função.
- Já para a adição dos *solvers* do *varFDTD* foi feito o seguinte código:

```

switchtolayout();

ATIVIDADE_CREATE;

x_span_dvc = 10e-6;
y_span_dvc = 8e-6;
solver_x_span = x_span_dvc + 1e-6;
solver_y_span = 2*y_span_dvc + 8e-6;
solver_z_span = 2e-6;
number_of_points = 50;
pontos = [[wg_radius/2, wg_radius/2];
[ wg_radius/2, -wg_radius/2];
[-wg_radius/2, wg_radius/2];
[-wg_radius/2, -wg_radius/2]];
# add solver
addvarfdtd();
set("x", -x_span_dvc/2);
set("y", -5e-6);
set("z", 0);
set("x span", solver_x_span);
set("y span", solver_y_span);
set("z span", solver_z_span);
set("x0", -5e-6);
set("test points", pontos);
set("background material", "SiO2 (Glass) - Palik");
set("global monitor frequency points", number_of_points);
set("mesh accuracy", 5);
# add source
addmodesource();
set("injection axis", "x");
set("x", -x_span_dvc-0.3e-6);
set("y", -5e-6);
set("y span", y_span_dvc);
set("wavelength start", 1550e-9);
set("wavelength stop", 1600e-9);
# add monitors
addindex();
set("name", "index");
set("x", -x_span_dvc/2);
set("y", -5e-6);
set("x span", solver_x_span);
set("y span", solver_y_span);
addprofile();
set("x", -x_span_dvc/2);
set("y", -5e-6);

```

```

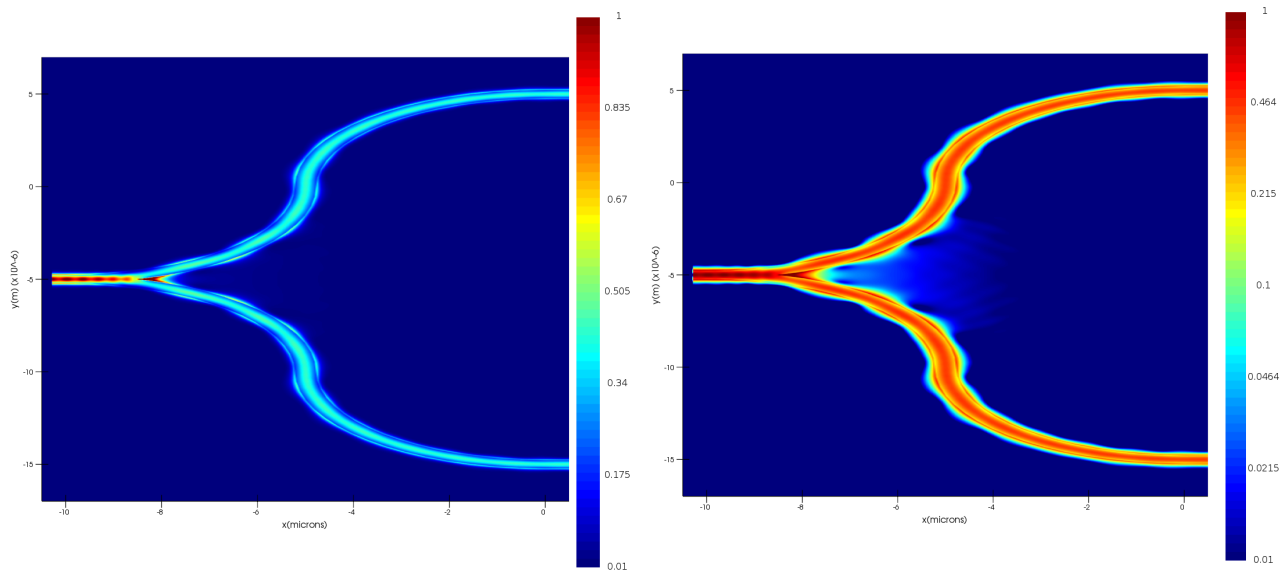
set("x span", solver_x_span);
set("y span", solver_y_span);
addpower();
set("name", "in");
set("monitor type", 5);
set("x", -10e-6);
set("y", -5e-6);
set("y span", y_span_dvc - 2e-6);
set("z span", solver_z_span);
addpower();
set("name", "out_1");
set("monitor type", 5);
set("x", x_span_dvc - 10e-6);
set("y", y_span_dvc - 3e-6);
set("y span", y_span_dvc - 2e-6);
set("z span", solver_z_span);
addpower();
set("name", "out_2");
set("monitor type", 5);
set("x", x_span_dvc - 10e-6);
set("y", -y_span_dvc - 7e-6);
set("y span", y_span_dvc - 2e-6);
set("z span", solver_z_span);

```

-
- Note que o texto “ATIVIDADE_CREATE” referencia o arquivo de mesmo nome que cria a estrutura, ou seja, cria a estrutura primeiro para depois adicionar os *solvers*. “addvarfdtd();” cria o solucionador *varFDTD* que será utilizado para obter os resultados. “addmodesource();” cria a fonte que injeta na direção x. Inicialmente tem-se apenas uma fonte, para que a estrutura se comporte como *splitter*. Posteriormente será adicionado nas portas de saída, fazendo se comportar como *combiner*. Posteriormente é acrescido apenas os monitores de perfil “addprofile();”, monitores de índices “addindex();” e monitores de campo “addpower();” (um para cada porta).

2.2. Simulação com *varFDTD*.

- O dispositivo foi simulado como *splitter*. Primeiramente foi obtido o perfil de campo em $||^2$. O resultado é ilustrado na Figura 2.

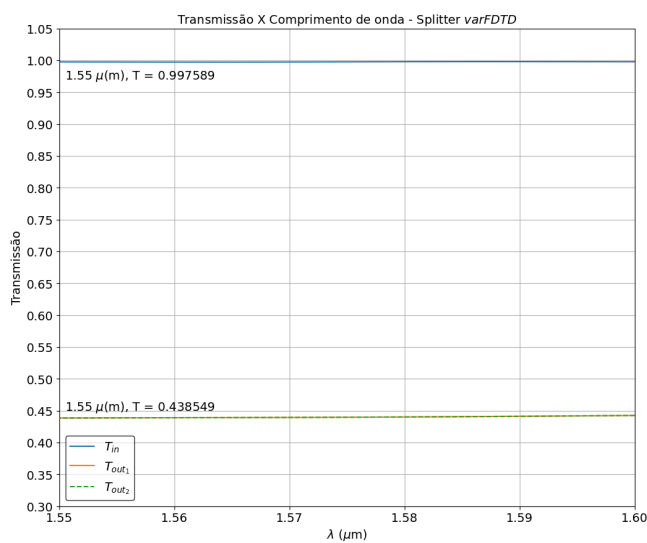


(a) Escala linear.

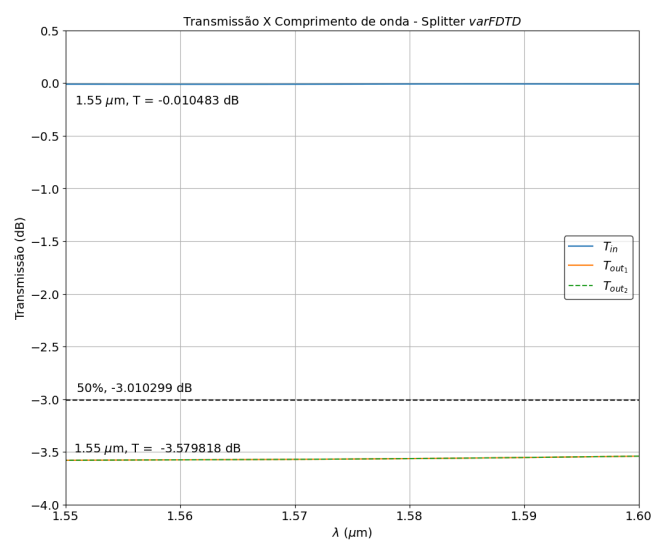
(b) Escalar log.

Figura 2 – Perfil de campo elétrico para *splitter*.

- Na Figura 3 é ilustrada a transmissão, tanto na escala linear quanto logarítmica.



(a) Escala linear.



(b) Escala Log.

Figura 3 – Transmissão para *splitter*.

- A transmissão em 1550 nm é igual a $-0,010483 \text{ dB}$ para a porta 1 (porta de entrada à esquerda).
- Para as portas de saída à direita (2 e 3) a transmissão obtida, em 1550 nm , é igual a $-3,579818 \text{ dB}$.
- A perda de inserção em 1550 nm de cada porta é igual a $0,569518 \text{ dB}$. Para as duas portas basta multiplicar por 2 obtendo assim $1,139037 \text{ dB}$.

- O dispositivo opera em toda a banda C. Conclui-se isso visto que, em toda a faixa analisada, a transmissão e a perda de inserção não se alteram, se comportando como esperado, ou seja, cada porta com uma transmissão próxima a -3 dB .
- Agora é realizada a simulação do dispositivo como *combiner*. Na Figura 4 é ilustrado o perfil de campo em $||^2$.

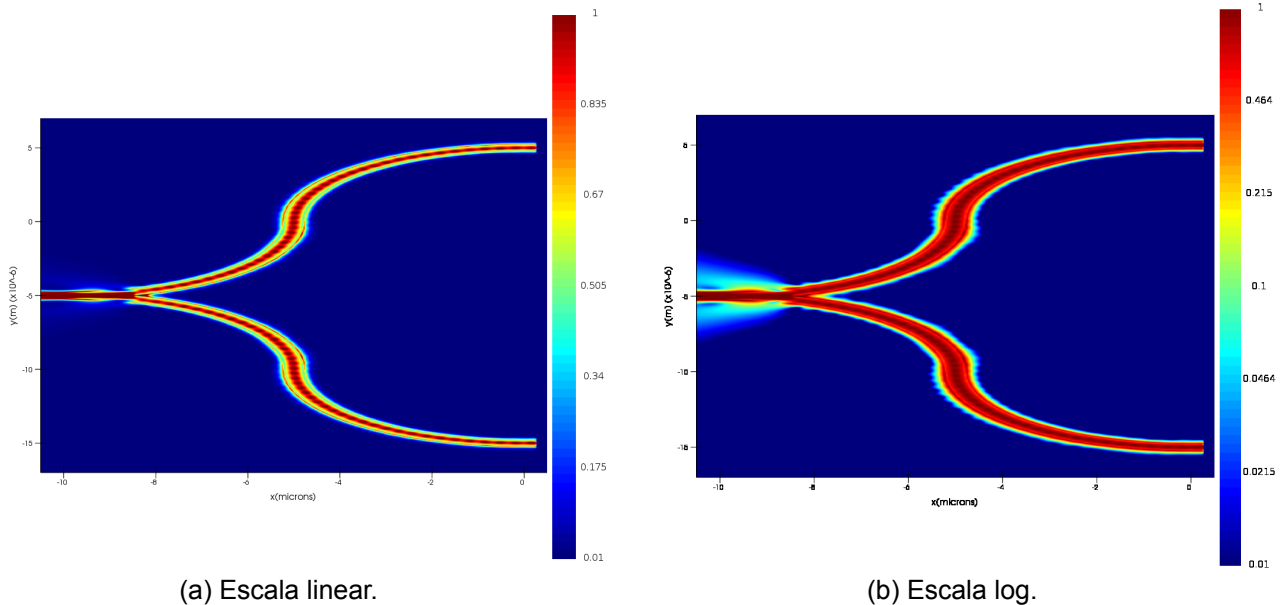


Figura 4 – Perfil de campo para *combiner*.

- Na Figura 5 é ilustrada a transmissão.

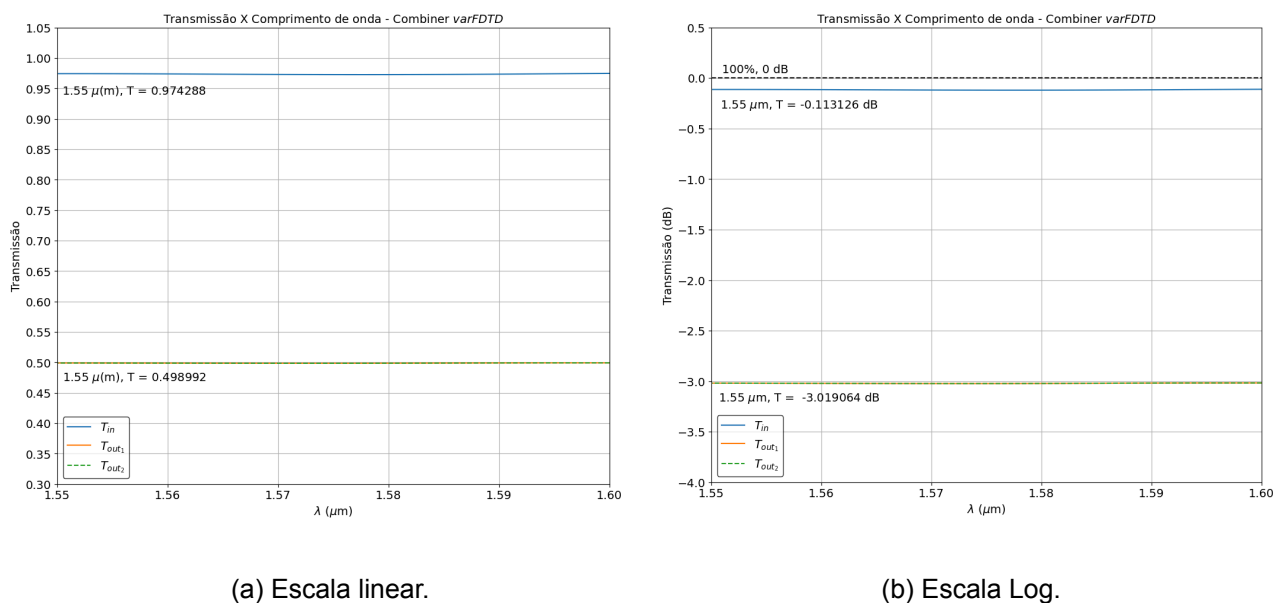


Figura 5 – Transmissão para *combiner*.

- A transmissão em 1550 nm é igual a $-0,113126\text{ dB}$ para a porta 2 e 3 (porta de saída à direita).

- Para a porta de entrada 1 à transmissão obtida, em 1550 nm , é igual a $-3,019064\text{ dB}$.
- A perda de inserção em 1550 nm de cada porta é igual a $0,113126\text{ dB}$ levando por base que o esperado na porta de entrada seria 0 dB , ou seja, um resultado próximo para uma estrutura não otimizada.
- O dispositivo opera em toda a banda C. O mesmo argumento apresentado anteriormente serve para essa situação, tomando por base que a transmissão basicamente não se altera na banda.

2.3. Simulação com FDTD.

2.3.1. Modo TE.

- Nessa etapa é visado a obtenção dos mesmos parâmetros (perfil de campo, transmissão e perda de inserção), porém agora utilizando o FDTD como *solver*. O código feito para a criação do *solver* encontra-se abaixo.

```

switchtolayout();
ATIVIDADE_CREATE;
x_span_dvc = 10e-6;
y_span_dvc = 8e-6;
solver_x_span = x_span_dvc + 1e-6;
solver_y_span = 2*y_span_dvc + 8e-6;
solver_z_span = 2e-6;
number_of_points = 50;
# add solver
addfDTD();
set("x", -x_span_dvc/2);
set("y", -5e-6);
set("z", 0);
set("x span", solver_x_span);
set("y span", solver_y_span);
set("z span", solver_z_span);
set("background material", "SiO2 (Glass) - Palik");
set("global monitor frequency points", number_of_points);
set("global source wavelength start", 1500e-9);
set("global source wavelength stop", 1600e-9);
set("mesh accuracy", 5);
set("y min bc", "Symmetric");

#add ports
addport();
set("name", "in");
set("x", -10e-6);
set("y", -5e-6);
set("y span", 2e-6);
set("z span", 2e-6);
addport();
set("name", "out 1");
set("x", 0);
set("y", 5e-6);
set("y span", 2e-6);
set("z span", 2e-6);
set("direction", "backward");
addport();
set("name", "out 2");

```



```

set("x", 0);
set("y", -15e-6);
set("y span", 2e-6);
set("z span", 2e-6);
set("direction", "backward");

#add monitors
addindex();
set("x", -x_span_dvc/2);
set("y", -5e-6);
set("x span", solver_x_span);
set("y span", solver_y_span);
addprofile();
set("x", -x_span_dvc/2);
set("y", -5e-6);
set("x span", solver_x_span);
set("y span", solver_y_span);
save("Y_branch__fdtd");

```

- Diferente do que foi apresentado para o *varFDTD*, não há a necessidade de pontos de teste. A função "addfdtd();" substitui a "addvarfdtd();". Nesse bloco vale destacar que, assim como no *varFDTD*, é escolhido o material da região vazia em preto. A função "background material" preenche o espaço vazio com o material escolhido, no caso SiO_2 . Além disso, a função "y min bc" seleciona tanto modo de simulação assimétrico ou simétrico, dependendo do modo ser TE ou TM, respectivamente. No lugar dos monitores "addpower();" e "addmodesource();" utiliza-se apenas "addport();".
- O dispositivo foi simulado como *splitter* para o modo TE. Primeiramente foi obtido o perfil de campo em $||^2$. O resultado é ilustrado na Figura 6.

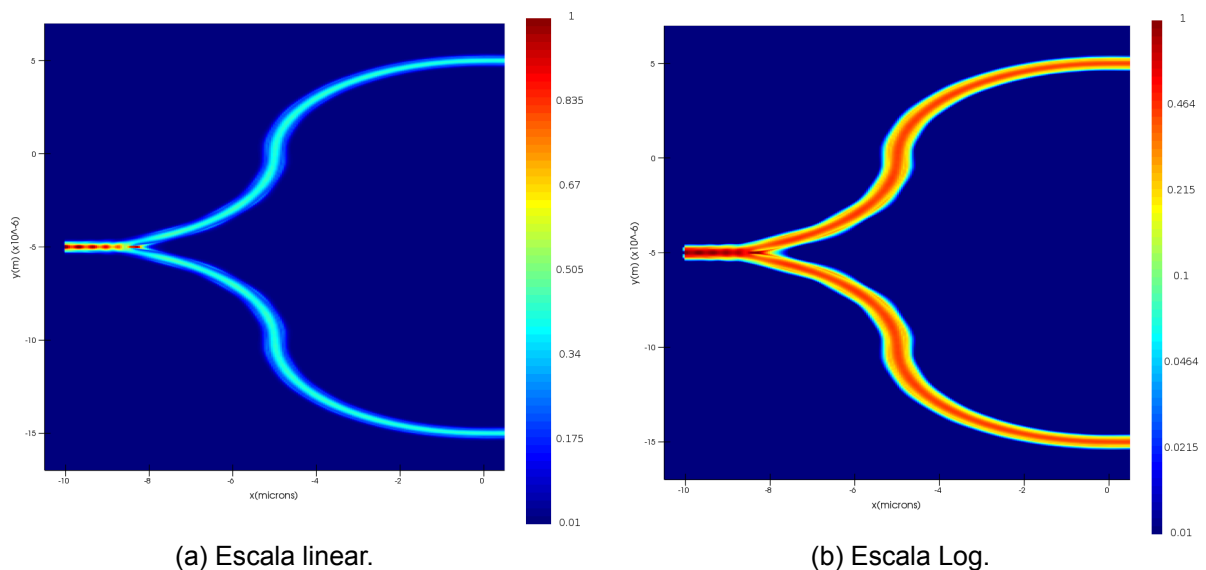


Figura 6 – Perfil de campo como *splitter* utilizando FDTD.

- Na Figura 7 é ilustrada a transmissão, tanto na escala linear quanto logarítmica.

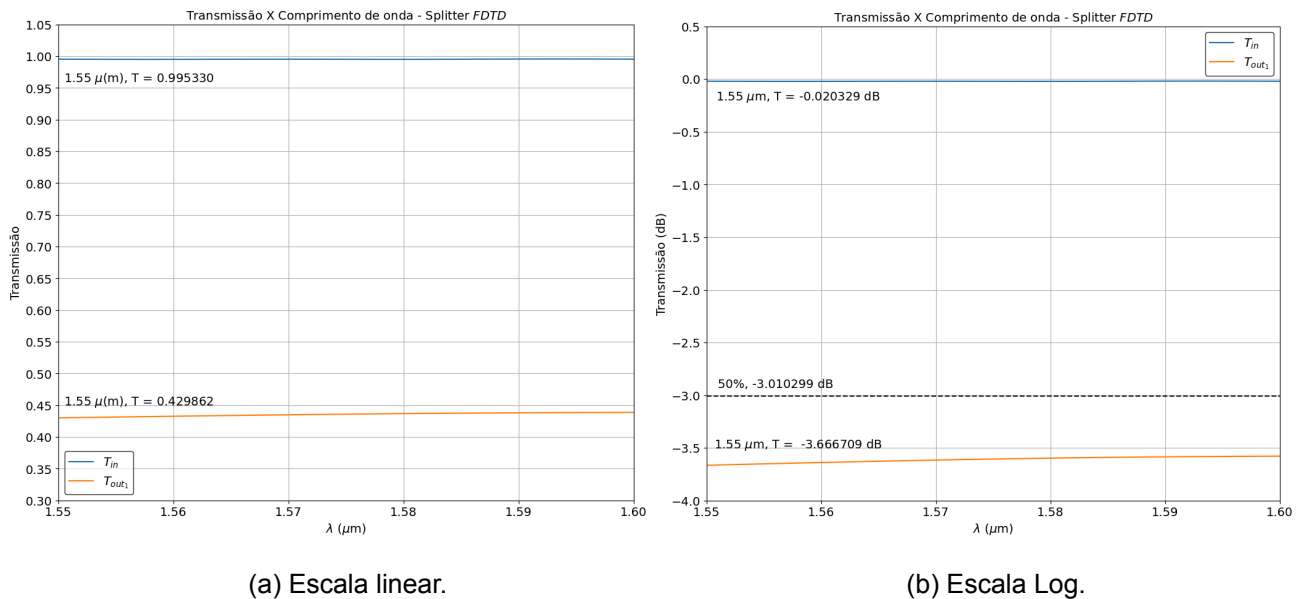
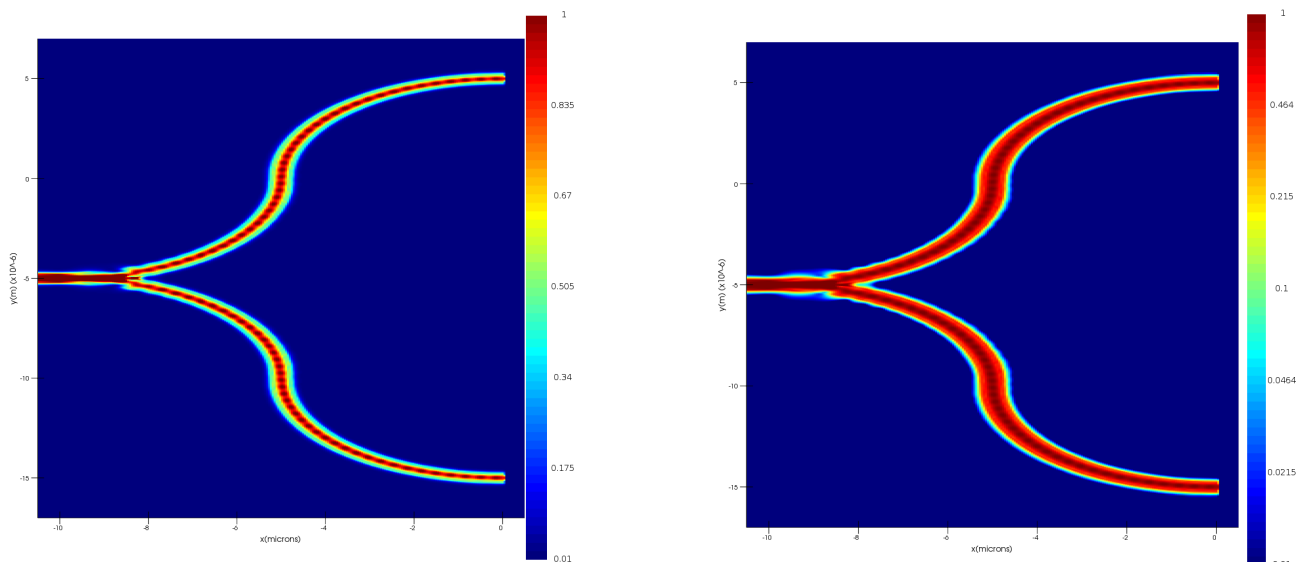


Figura 7 – Transmissão para *splitter* utilizando FDTD.

- A transmissão em 1550 nm é igual a $-0,020329 \text{ dB}$ para a porta 1 (porta de entrada à esquerda).
- Para as portas de saída à direita (2 e 3) a transmissão obtida, em 1550 nm , é igual a $-3,666709 \text{ dB}$.
- A perda de inserção em 1550 nm de cada porta é igual a $0,656409 \text{ dB}$. Para as duas portas basta multiplicar por 2 obtendo assim $1,312818 \text{ dB}$.
- Comparando a Figura 7 com a Figura 3 nota-se uma pequena discrepância entre os valores de transmissão em 1550 nm . O posicionamento dos monitores são distintos em ambos os ambientes de simulação, logo é esperado essa discrepância. A diferença percentual entre eles é igual a $1,98\%$.
- Agora é realizada a simulação do dispositivo como *combiner*, também modo TE. Na Figura 8 é ilustrado o perfil de campo em $||^2$.

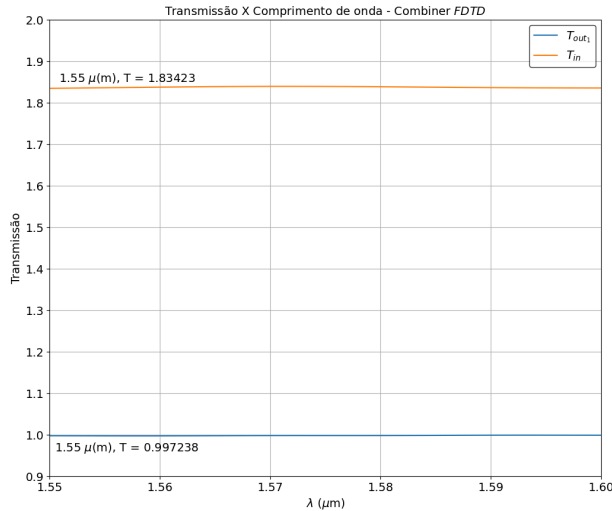


(a) Escala linear.

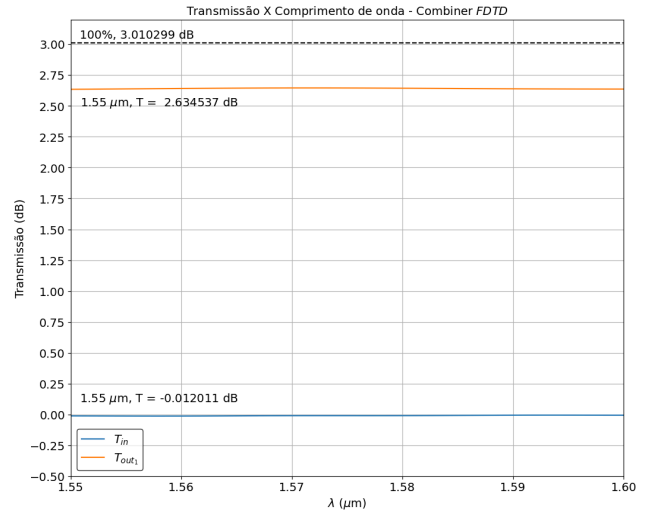
(b) Escala Log.

Figura 8 – Perfil de campo como *combiner* utilizando FDTD.

- Na Figura 9 é ilustrada a transmissão.



(a) Escala linear.



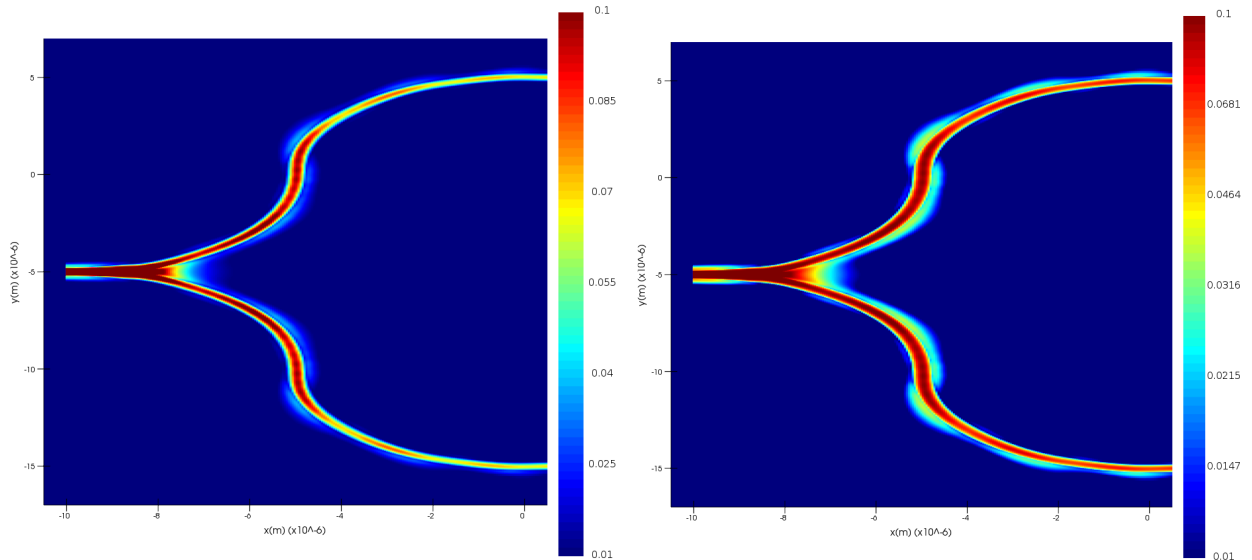
(b) Escala Log.

Figura 9 – Transmissão para *combiner* utilizando FDTD.

- A transmissão em 1550 nm é igual a $-0,012011 \text{ dB}$ para a porta 2 e 3 (porta de saída à direita).
- Para a porta de entrada 1 (à esquerda) a transmissão obtida, em 1550 nm , é igual a $2,634537 \text{ dB}$.
- A perda de inserção em 1550 nm é igual a $0,375762 \text{ dB}$.
- Nesse caso há uma discrepância maior entre os valores, mas isso pode ser explicado apenas por questão referencial. Note pela Figura 5 que a transmissão nas portas à direita são iguais a 0,5, aproximadamente, enquanto que a transmissão na simulação via FDTD possui transmissão igual a 1, aproximadamente. É esperado, na porta à esquerda, que a transmissão seja igual ao dobro da transmissão da entrada de uma porta individual. Em ambos os casos isso ocorre, mas, por estar sendo utilizada as funções “Symmetric” e “Anti-Symmetric”, ocorre a simulação apenas em uma das portas à direita, visto que existe simetria, adotando assim transmissão unitária, aproximadamente.

2.3.2. Modo TM.

- Para as simulações em modo TM muda-se “y min bc” para Symmetric.
- O dispositivo foi simulado como *splitter* para o modo TM. Primeiramente foi obtido o perfil de campo em $||^2$. O resultado é ilustrado na Figura 10.

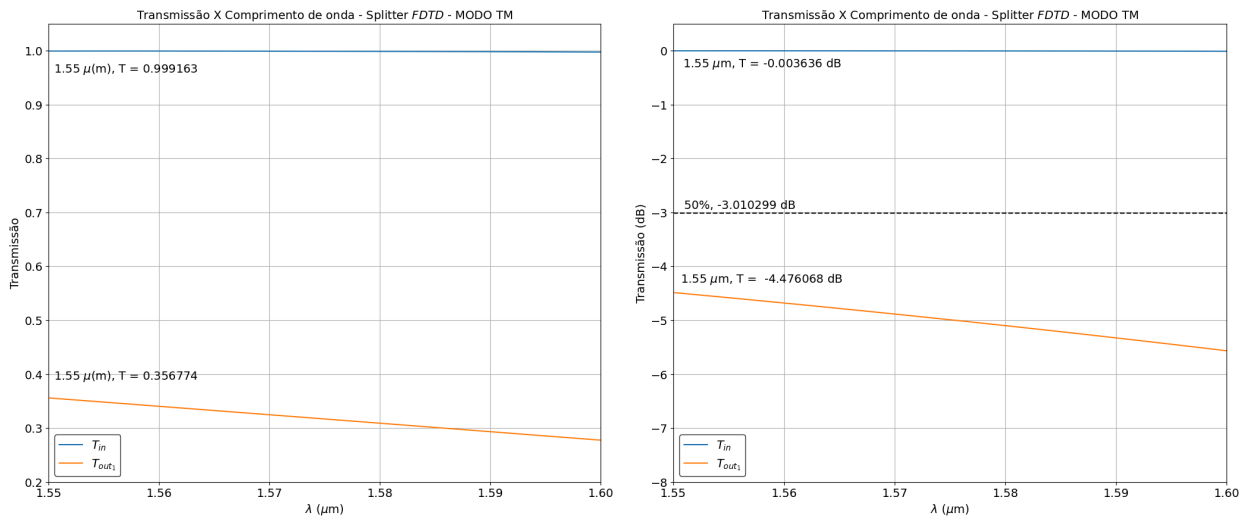


(a) Escala linear.

(b) Escala Log.

Figura 10 – Perfil de campo como *splitter* utilizando FDTD, para modo TM.

- Na Figura 11 é ilustrada a transmissão, tanto na escala linear quanto logarítmica.



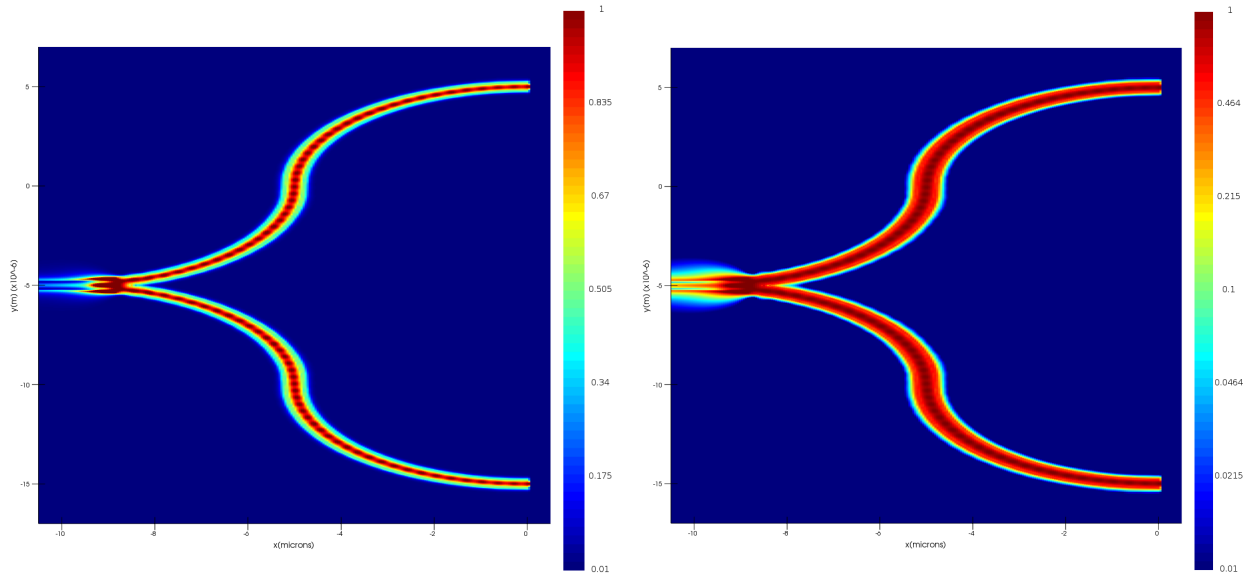
(a) Escala linear.

(b) Escala Log.

Figura 11 – Transmissão para *splitter* utilizando FDTD, para modo TM.

- A transmissão em 1550 nm é igual a $-0,003636 \text{ dB}$ para a porta 1 (porta de entrada à esquerda).

- Para as portas de saída à direita (2 e 3) a transmissão obtida, em 1550 nm , é igual a $-4,476068\text{ dB}$.
- A perda de inserção em 1550 nm de cada porta é igual a $1,465768\text{ dB}$. Para as duas portas basta multiplicar por 2 obtendo assim $2,931536\text{ dB}$.
- Agora é realizada a simulação do dispositivo como *combiner* para o modo TM. Na Figura 12 é ilustrado o perfil de campo em $||^2$.

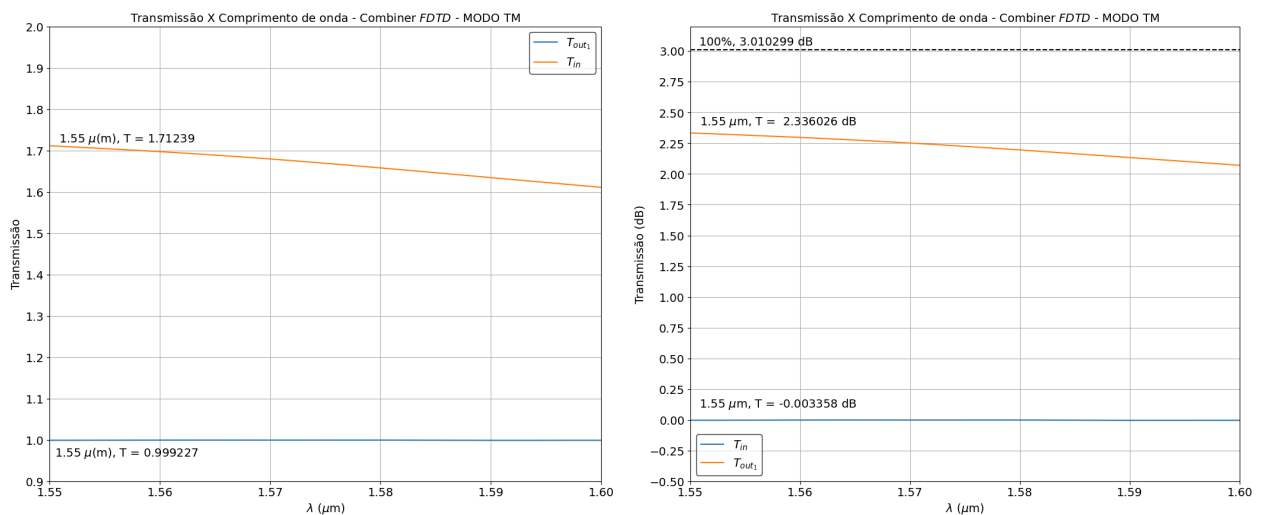


(a) Escala linear.

(b) Escala Log.

Figura 12 – Perfil de campo como *combiner* utilizando FDTD, para modo TM.

- Na Figura 13 é ilustrada a transmissão.



(a) Escala linear.

(b) Escala Log.

Figura 13 – Transmissão para *combiner* utilizando FDTD, para modo TM.

- A transmissão em 1550 nm é igual a $-0,003358\text{ dB}$ para a porta 2 e 3 (porta de saída à direita).

- Para a porta de entrada 1 (à esquerda) a transmissão obtida, em 1550 nm , é igual a $2,336026\text{ dB}$.
- A perda de inserção em 1550 nm é igual a $0,674273\text{ dB}$.
- Uma consideração que pode ser realizada sobre as simulações para modo TM é a respeito dos gráficos que apresentam uma característica menos linear se comparada às simulações feitas para o modo TE. Isso ocorre, visto que o modo fundamental propagado é TE logo é mais confinado. O modo TM, por outro lado, é menos confinado, logo, dependendo da região da banda C, a perda de inserção é bem mais elevada.

2.4. Simulação via **INTERCONNECT**.

- Por esse método é possível obter os parâmetros de espalhamento S e assim exportá-los para o *interconnect*. Na Figura 14 é ilustrado o resultado obtido dos parâmetros S, para o modo TE, por meio do FDTD e operando como *splitter*.

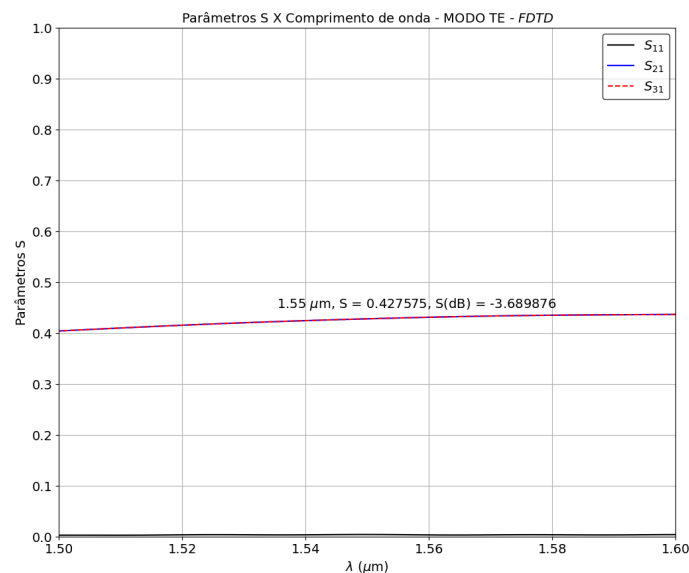


Figura 14 - Parâmetros S via simulação com FDTD.

- Os valores obtidos para o parâmetros S_{11} são próximo a 0, logo há pouca reflexão, como esperado.
- Já a Figura 15 representa os parâmetros S obtidos por meio do interconnect. A Figura 16 mostra o circuito fotônico desenvolvido para se obter esses parâmetros S.

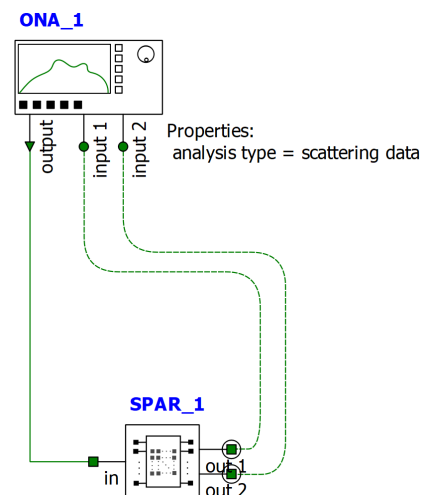
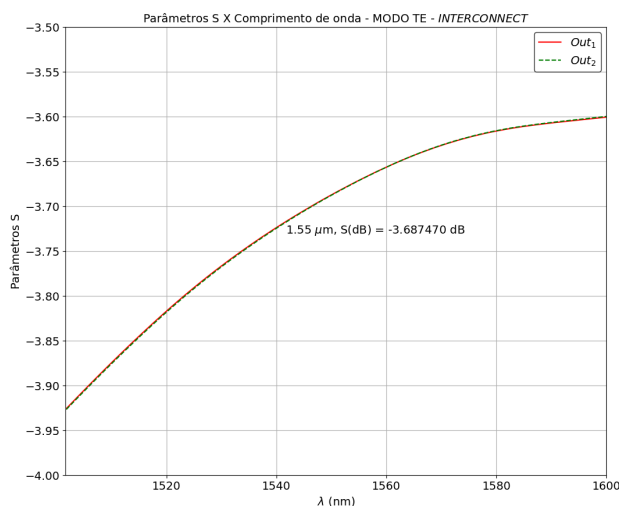


Figura 15 – Parâmetros S.

Figura 16 – Modelo de simulação.

- Note pelos valores marcados no gráfico que existe um erro mínimo igual a 0,0652% entre as simulações via *FDTD* e *interconnect*. Por padrão o *interconnect* retorna os parâmetros (em dB) apenas nas saídas, nesse caso, porta 2 e 3 à direita (out_1 e out_2).
- Agora, a mesma estrutura, ainda operando como *splitter* e modo TM, apresentou o resultado ilustrado pela Figura 16, considerando que foi utilizado o *FDTD*.

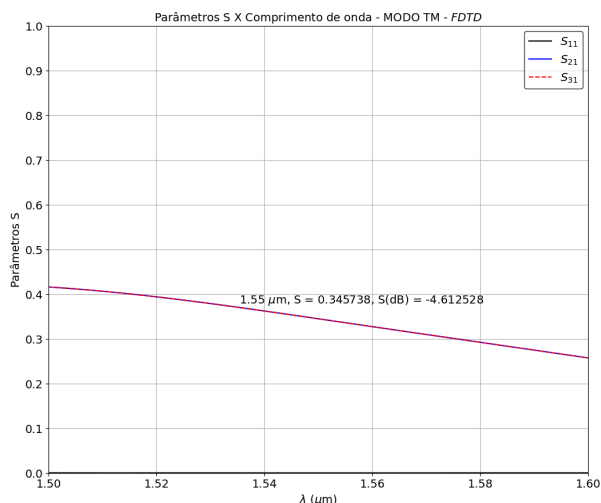


Figura 16 – Parâmetros S.

- Já os resultados via *interconnect* são ilustrados na Figura 17.

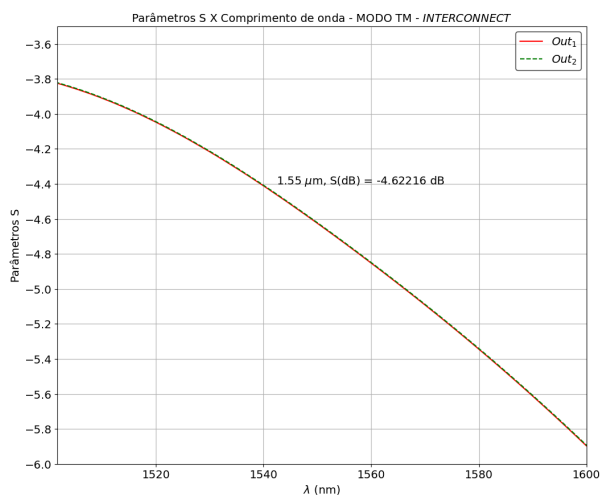


Figura 17 – Parâmetros S.

- Comparando com os resultados simulados via *FDTD*, os parâmetros *S* possuem comportamento semelhante ao que é visto nos gráficos das transmissões, especialmente para o modo TE. O modo TM tem uma grande discrepância quando comparado ao que foi simulado via *interconnect*.