

Universidade Federal de São Carlos – UFSCar
Centro de Ciências Exatas e de Tecnologia – CCET
Departamento de Física – DF

Douglas Ramos do Amaral

**Software para análise de dados experimentais de uma técnica
de medida baseada em ressonadores Fabry-Pérot**



São Carlos
2023

Douglas Ramos do Amaral

Software para análise de dados experimentais de uma técnica de medida baseada em ressonadores Fabry-Pérot

Monografia apresentada à Banca Examinadora e disciplina 97624 - Trabalho Final de Curso do Departamento de Física da Universidade Federal de São Carlos como requisito para a obtenção do grau de bacharel em Engenharia Física.

Orientador: Prof. Dr. Filippo Giovanni Ghiglieno

São Carlos

2023

LISTA DE FIGURAS

Figura 1 – Diagrama de um etalon que constitui um interferômetro de Fabry-Pérot. A luz entra no etalon e sofre múltiplas reflexões internas. Os Rs são os feixes transmitidos de volta para o mesmo lado do feixe incidente, enquanto os Ts são os feixes transmitidos para o espaço subsequente ao etalon. l e n representam a espessura e índice de refração do meio respectivamente. . . .	7
Figura 2 – Cella unitária do IFP de múltiplas cavidades. E_i são as amplitudes do campo elétrico dos feixes nas superfícies semi-refletoras e l_i é a distância entre as superfícies i e $i + 1$	8
Figura 3 – Resultados de perfis de transmissão de IFP de duas, três e quatro cavidades (a) modelados a partir das equações analíticas (b) obtidos através de medidas experimentais normalizadas. 2C, 3C e 4C são as curvas para 2, 3 e 4 cavidades respectivamente. As distâncias assinaladas representam o <i>free spectral range</i> (FSR) que é a distância entre dois picos consecutivos.	9
Figura 4 – Diagrama esquemático do IFP de quatro espelhos. L_1 , L_2 e L_3 são os três comprimentos das cavidades. Os espelhos 1 e 4 são fixos, enquanto os espelhos 2 e 3 são móveis com deflexões $\Delta 1$ e $\Delta 2$ respectivamente nas direções indicadas pelas setas.	10
Figura 5 – Resposta espectral do IFP quando sintonizado desviando o espelho 2 em $\Delta 1$ e mantendo $\Delta 2$ em zero para todas as respostas.	10
Figura 6 – Sistema óptico de N camadas dielétricas e resultados da simulação do sistema feita no software MATLAB.	11
Figura 7 – Aparato experimental da técnica para extrair as perdas por espalhamento. . .	13
Figura 8 – Região da guia de onda (amostra) com as três cavidades evidenciadas. . . .	14
Figura 9 – Comportamento da propagação de um laser em um etalon.	15
Figura 10 – Representação do traçado de raios evidenciando a diferença de caminho óptico entre dois feixes consecutivos.	15
Figura 11 – Representação esquemática de um IFP de N interfaces refletoras.	17
Figura 12 – Layout do IDE Spyder.	25
Figura 13 – Layout do programa desenvolvido para simular o comportamento da Transmissão T em função de λ para um IFP de 3 cavidades.	27
Figura 14 – Gráfico da transmitância T em função de λ para diferentes valores de R_1 e R_4 . . .	28
Figura 15 – Espectro de transmissão para diferentes valores de l_1 e l_3 indicados nos gráficos. No gráfico superior variou-se apenas l_1 e no gráfico do centro variou-se l_3 . Já no gráfico inferior foram variados ambos os parâmetros. . .	29
Figura 16 – Gráficos do espectro de transmissão para três valores distintos do comprimento óptico da cavidade 2 ($n_2 l_2$).	30

Figura 17 – Software para análise dos dados experimentais da técnica de perdas por espalhamento baseada em IFP de três cavidades.	30
Figura 18 –	32

LISTA DE TABELAS

Tabela 1 – Informações das medidas presentes em cada arquivo de dados.	31
Tabela 2 – Resultados dos parâmetros do IFP de três cavidades inferidos pela análise dos dados feita através do software desenvolvido no trabalho.	32

SUMÁRIO

1	INTRODUÇÃO	6
2	REVISÃO BIBLIOGRÁFICA	8
3	APARATO EXPERIMENTAL DA TÉCNICA PARA EXTRAIR AS PERDAS POR ESPALHAMENTO	13
4	FUNDAMENTAÇÃO TEÓRICA	15
4.1	Interferômetro de Fabry-Perot (IFP)	15
4.2	Interferômetro de Fabry-Perot de 3 cavidades	19
5	FERRAMENTAS E MÉTODOS	25
5.1	Ferramentas	25
5.2	Métodos	26
6	RESULTADOS E DISCUSSÃO	27
6.1	Software de Simulação	27
6.1.1	Funcionamento	27
6.1.2	Simulação	28
6.2	Software de Análise de dados	29
6.2.1	Funcionamento	29
6.2.2	Análise dos dados	31
7	CONCLUSÃO	33
	REFERÊNCIAS	34
	APÊNDICE A – CÓDIGO FONTE DO PROGRAMA DE SIMULAÇÃO	37
	APÊNDICE B – CÓDIGO FONTE DO SOFTWARE DE ANÁLISE DOS DADOS EXPERIMENTAIS	44
	ANEXO A – TÍTULO DO ANEXO A	55
	ANEXO B – TÍTULO DO ANEXO B	56
	ANEXO C – TÍTULO DO ANEXO C	57

1 INTRODUÇÃO

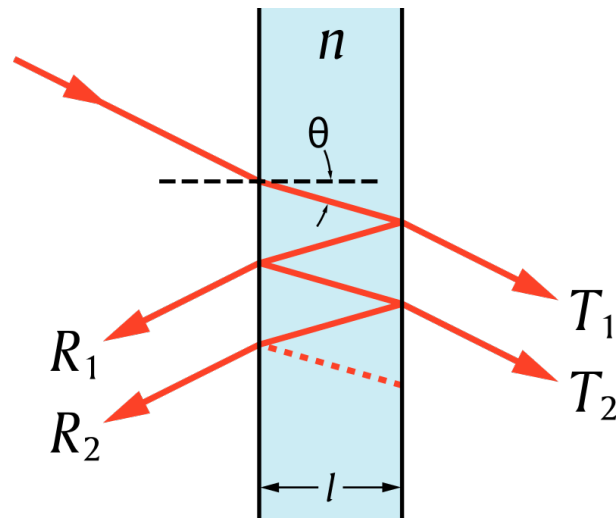
A fotônica compreende o estudo científico e a aplicação da luz com base na geração, manipulação e detecção de fótons e ondas eletromagnéticas (CULSHAW, 2020). Todo e qualquer dispositivo eletrônico que envolva fótons em sua operação está encerrado na fotônica. Atualmente, essa área é onipresente, estando em todos os ramos da ciência, na pesquisa e desenvolvimento de tecnologias de ponta (PELUCCHI et al., 2021; SINGH; DAS; RAY, 2023), aplicações industriais e, claro, no cotidiano das pessoas com os aparelhos pessoais e utilidades domésticas. Dentre as mais diversas aplicações compreendidas na fotônica, destacam-se: detecção de luz, telecomunicações, processamento de informações, energia fotovoltaica, computação fotônica e quântica, iluminação, metrologia, halografia, espectroscopia, diagnóstico e terapia médicas, detecção biológica e química (CHANDRASEKAR et al., 2019), agricultura (TAN et al., 2019) e robótica. Do ponto de vista econômico e social, a fotônica representa uma indústria global avaliada em 10,2 bilhões de dólares em 2022 (TRANSPARENCY MARKET RESEARCH, INC., 2023), sendo imprescindível no modo de vida tecnológico atual da sociedade.

As aplicações da fotônica são viabilizadas por dispositivos que utilizam materiais semicondutores passivos e ativos para geração, amplificação e manipulação da luz. Um critério muito importante que afeta o ótimo desempenho desses materiais são as perdas por espalhamento, que devem ser mínimas. A luz nos dispositivos fotônicos representa eficiência e informação e não é desejado que tais parâmetros sejam prejudicados por materiais que, na verdade, deveriam promovê-los. Nesse sentido, é crucial a caracterização dos materiais com elevado potencial de aplicação nos dispositivos fotônicos para analisar suas perdas por espalhamento e definir se estes apresentam um desempenho satisfatório para o uso em tais dispositivos.

Levando em conta a importância dessa caracterização para o desenvolvimento de novos dispositivos, O Grupo de Óptica, Laser e Fotônica (OLAF) do Departamento de Física (DF) da Universidade Federal de São Carlos (UFSCar) implementou uma técnica simples e barata para realizar medidas de perdas por espelhamento de materiais passivos aplicáveis a fotônica, como guias de onda para óptica não-linear e nanofios. O sistema de caracterização é baseado em propriedades provenientes de um mecanismo físico denominado Interferômetro de Fabry-Pérot (IFP).

Um IFP consiste, normalmente, de uma cavidade formada por duas superfícies parcialmente refletoras, opticamente planas e paralelas (FOWLES, 2012), conforme é ilustrado na Figura 1. Utilizando o princípio de interferência de múltiplos feixes, o IFP funciona como um filtro, permitindo a transmissão de feixes de luz com comprimentos de onda (λ) específicos, determinados por alguns parâmetros da cavidade. Quando as superfícies são mantidas a uma distância fixa e entre elas há um material com índice de refração n , o Interferômetro de Fabry-Pérot recebe o nome de etalon.

Figura 1 – Diagrama de um etalon que constitui um interferômetro de Fabry-Pérot. A luz entra no etalon e sofre múltiplas reflexões internas. Os Rs são os feixes transmitidos de volta para o mesmo lado do feixe incidente, enquanto os Ts são os feixes transmitidos para o espaço subsequente ao etalon. l e n representam a espessura e índice de refração do meio respectivamente.



Fonte: Wikimedia Commons (2009).

O aparato experimental da técnica desenvolvida pelo OLAF constitui um sistema de três cavidades de Fabry-Pérot, ou seja, a transmissão da luz, que é a informação obtida pelo experimento, é determinada por um IFP de quatro interfaces. A técnica de medição das perdas é prejudicada pelas cavidades a mais presentes no experimento. Felizmente, tal sistema, mesmo com mais de uma cavidade, possui uma solução analítica, embora ainda de difícil análise por conta do elevado número de parâmetros e termos. Uma simulação computacional utilizando tal solução permitiria uma melhor compreensão do fenômeno físico da interferência de múltiplas reflexões e evidenciaria a interdependência entre os diferentes parâmetros e como estes definem o comportamento da transmissão da luz para diferentes comprimentos de onda. Comparando-se os dados empíricos da transmissão com os dados obtidos pela equação analítica, pode-se determinar as características do IFP: os comprimentos das cavidades ópticas e refletividades das superfícies. Tais informações permitem que sejam feitas correções no aparato experimental a fim de eliminar as cavidades de Fabry-Pérot extras, deixando apenas a cavidade de interesse cujo o meio é o material a ser analisado.

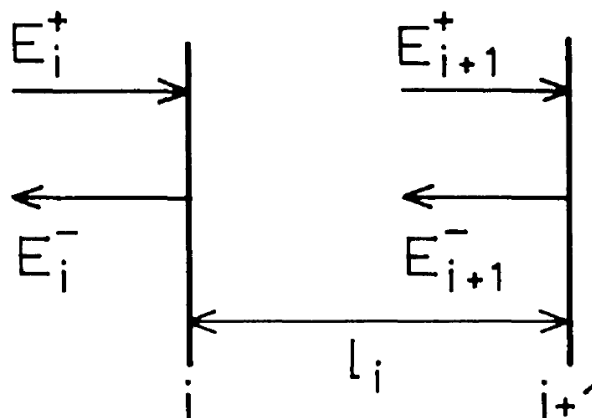
O objetivo deste trabalho consistiu em desenvolver um programa de computador que automatizasse a análise dos dados da técnica de medição das perdas por espelhamento, identificando de forma quantitativa os parâmetros do Interferômetro de Fabry-Pérot que constitui o sistema de medição, a saber: os comprimentos das cavidades e refletividades das superfícies. Deste modo, pode-se reconhecer quais elementos do aparato experimental estão produzindo cavidades de Fabry-Pérot indesejadas para posterior correção e eliminação das mesma do sistema.

2 REVISÃO BIBLIOGRÁFICA

Desde o momento em que Fabry e Perot apresentaram seu relato inaugural (FABRY; PÉROT, 1897), o Interferômetro de Fabry-Pérot (IFP) tem desempenhado uma função extremamente significativa em múltiplos âmbitos da óptica e fotônica (PFEIFER et al., 2022) que incluem lasers (CORZINE et al., 1989; KIM; KANG; LE, 2000; JAVALOYES; BALLE, 2010), filtros (ATHERTON et al., 1981; IRMER et al., 2003; MÉNARD; KIRK, 2009) e sensores (ISLAM et al., 2014). Muitos trabalhos foram desenvolvidos para vislumbrar novas aplicações e explorar as funcionalidades do IFP de cavidade única e mais ainda do IFP de múltiplas cavidades.

O trabalho de Stadt e Muller (1985) foi muito importante para estabelecer o comportamento físico do IFP de múltiplas cavidades e serviu como base para diversas pesquisas posteriores. Os autores utilizaram um método de cálculo matricial para determinar a relação geral para um IFP com um número N de cavidades ópticas. Além disso, foram apresentadas as equações analíticas da transmissão da luz para o interferômetro de uma, duas e três cavidades. Como mostra a Figura 2, Stadt e Muller idealizaram um IFP simples de superfícies semi-refletores paralelas e distância interplanar l_i como elemento óptico e associaram uma matriz de transformação (matriz ABCD). O IFP simples é a cela unitária de qualquer IFP de múltiplas cavidades, logo, a matriz ABCD associada a um IFP com uma quantidade N de cavidades é obtida multiplicando a matriz da cela unitária N vezes. Determinadas as matrizes para N de 1 a 3, os autores explicitaram as equações para cada caso e construíram gráficos da transmitância do IFP em função da diferença de fase entre os feixes transmitidos.

Figura 2 – Cela unitária do IFP de múltiplas cavidades. E_i são as amplitudes do campo elétrico dos feixes nas superfícies semi-refletores e l_i é a distância entre as superfícies i e $i + 1$.

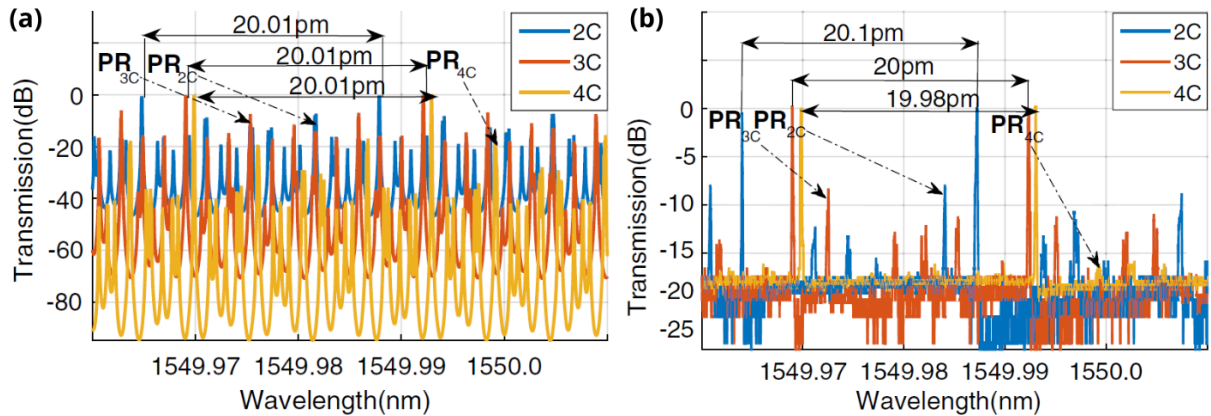


Fonte: Stadt e Muller (1985).

De forma similar, no artigo de Iftikhar, Khan e Cheema (2020) é apresentado o desenvolvimento matemático pelo método matricial para definir as equações da transmitância (T) de etalons de múltiplas cavidades. Os autores, a partir das relações obtidas, simularam o comportamento de T para sistemas de 2, 3 e 4 cavidades e compararam com dados experimentais,

conforme mostra a Figura 3.

Figura 3 – Resultados de perfis de transmissão de IFP de duas, três e quatro cavidades (a) modelados a partir das equações analíticas (b) obtidos através de medidas experimentais normalizadas. 2C, 3C e 4C são as curvas para 2, 3 e 4 cavidades respectivamente. As distâncias assinaladas representam o *free spectral range* (FSR) que é a distância entre dois picos consecutivos.

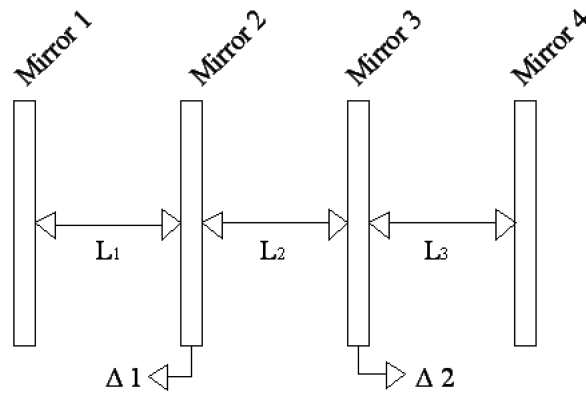


Fonte: Liu, Ukhtary e Saito (2017).

Márquez-Cruz e Hernández-Cordero (2014) examinaram os sinais de interferência refletidos de uma gota na extremidade clivada de uma fibra óptica visando estabelecer uma nova técnica para analisar a tensão superficial de líquidos. Os pesquisadores ajustaram os padrões de interferência a um modelo Fabry-Pérot de múltiplos espelhos. Mais uma vez, foi adotado o formalismo matricial para deduzir as equações de um IFP contendo um número N de espelhos. As expressões foram utilizadas para fazer os ajustes dos dados experimentais e determinar os parâmetros da gota suspensa na fibra e, assim, os aspectos de viscosidade.

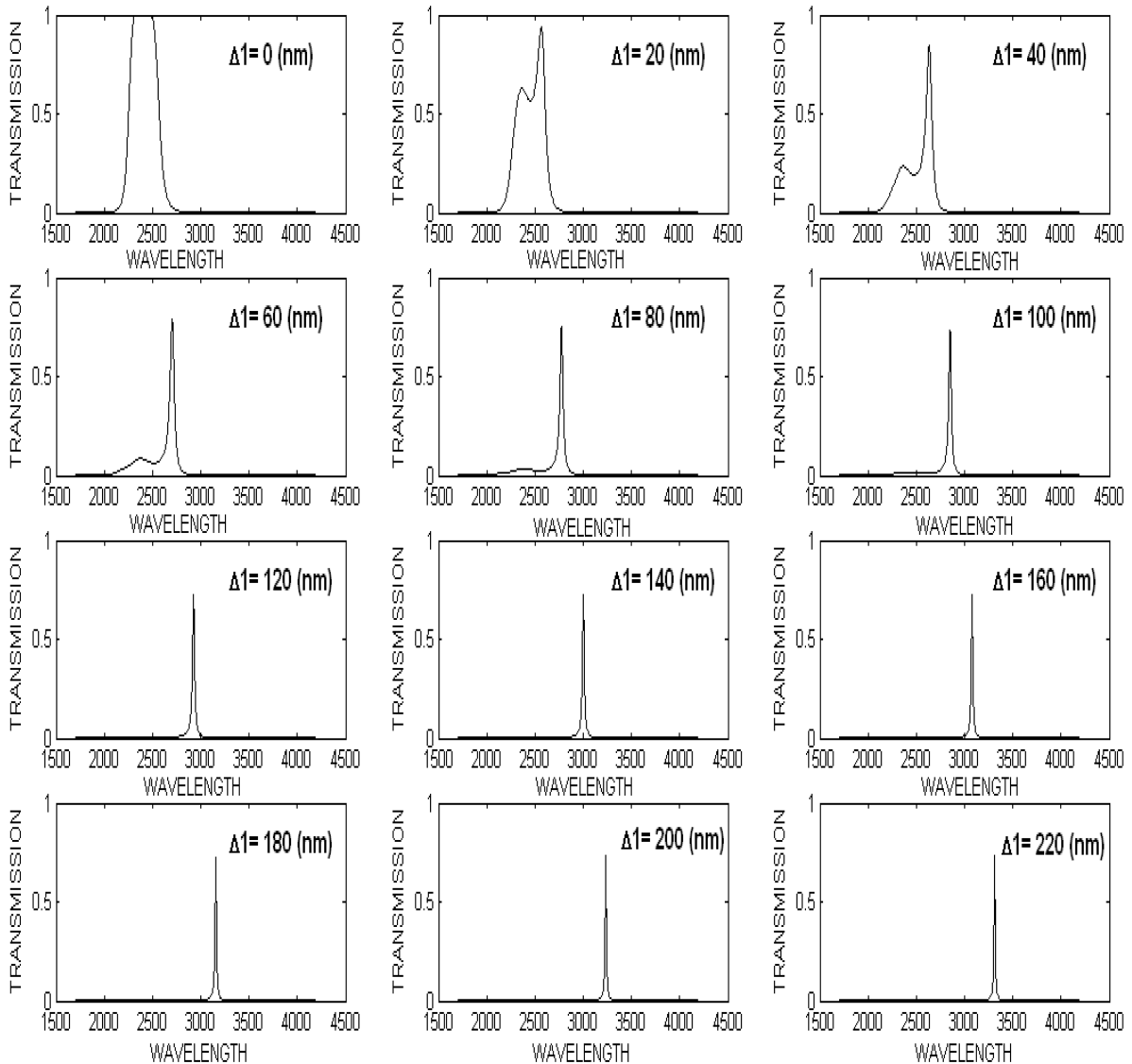
Outro trabalho interessante foi realizado por Parashar et al. (2007), em que um conceito de IFP de quatro espelhos (ou três cavidades) sintonizáveis, conforme ilustrado na Figura 4, foi elaborado como um sistema microeletromecânico (MEMS - *micro-electro-mechanical systems*) a ser aplicado como sensor. Um dispositivo constituído de um Interferômetro de Fabry-Pérot de quatro espelhos possui menos perdas de transmissão e uma maior faixa espectral de operação na região infravermelho do que dispositivos de arranjos em série de IFP de dois espelhos. Os autores utilizaram o método matricial para construir o modelo matemático do IFP de quatro espelhos. Foi definido um IFP de cavidade única como elemento óptico unitário, determinada sua matriz de transformação e replicada três vezes para obter a matriz do IFP múltiplo. Em seguida, os elementos da matriz foram utilizados para expressar as equações da intensidade de transmissão da luz que atravessa o IFP. O modelo desenvolvido pelos autores possibilitou simular o comportamento da intensidade luminosa transmitida em função do comprimento de onda da luz incidente para diferentes configurações dos comprimentos das cavidades ópticas do IFP. Alguns gráficos gerados a partir da simulação do IFP de quatro espelhos são mostrados na Figura 5, em que os comprimentos L_1 e L_2 das cavidades são alterados pelo deslocamento do espelho 2 (variação do $\Delta 1$). A partir deles é obtida a faixa espectral e a sensibilidade alcançada pelo sistema óptico.

Figura 4 – Diagrama esquemático do IFP de quatro espelhos. L_1 , L_2 e L_3 são os três comprimentos das cavidades. Os espelhos 1 e 4 são fixos, enquanto os espelhos 2 e 3 são móveis com deflexões $\Delta 1$ e $\Delta 2$ respectivamente nas direções indicadas pelas setas.



Fonte: Parashar et al. (2007).

Figura 5 – Resposta espectral do IFP quando sintonizado desviando o espelho 2 em $\Delta 1$ e mantendo $\Delta 2$ em zero para todas as respostas.



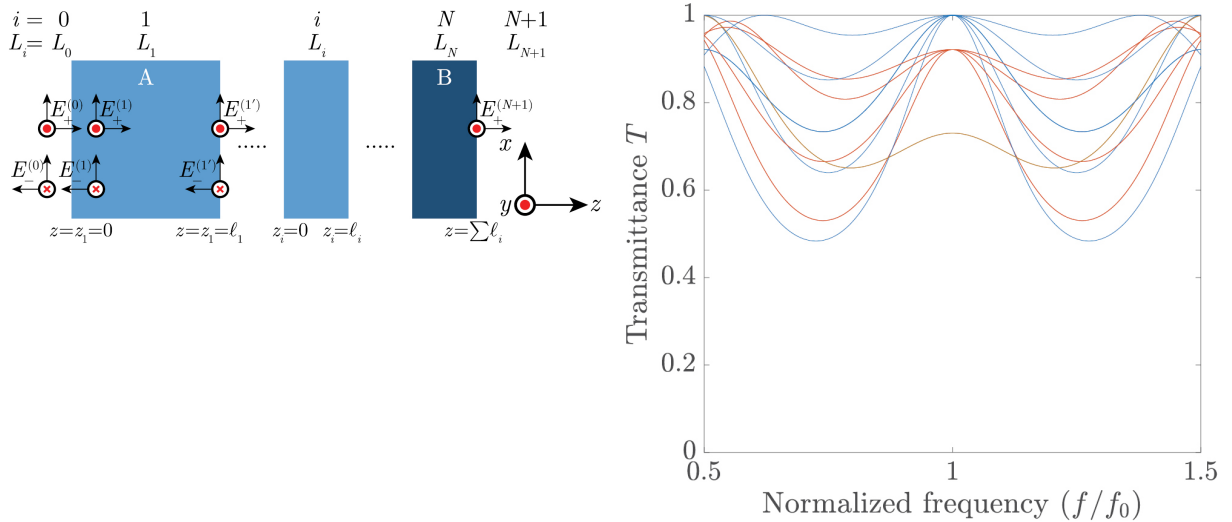
Fonte: Parashar et al. (2007).

O método matricial para modelar o IFP de quatro interfaces também foi utilizado na pesquisa de Costa et al. (2021), cujo objetivo foi investigar a imagem contínua em terahertz (CW THz - *continuous wave terahertz*) em um nível fundamental. O sistema de imagem utilizado foi um C-scan simples que configura um IFP de três cavidades. Os autores incluíram nos cálculos fatores dissipativos de absorção e dispersão do feixe representados por uma constante atribuída para cada cavidade.

Liu, Ukhtary e Saito (2017) fizeram um estudo sobre as propriedades ópticas de um sistema de N camadas arbitrárias de dielétricos. Dois tipos de materiais foram considerados para cada camada. A Figura 6a apresenta uma representação do sistema adotado pelos autores. Um arranjo em série de materiais dielétricos caracteriza um ressonador Fabry-Pérot. Esse mecanismo é empregado como filtro óptico, uma vez que permite o controle de propriedades da propagação da luz que atravessa o conjunto de camadas dielétricas, como espectro de transmissão, velocidade de grupo e dispersão. Os autores serviram-se do método de matriz de transferência para calcular a transmitância do sistema para qualquer sequência arbitrária de meios dielétricos. Utilizando o software MATLAB, Liu, Ukhtary e Saito (2017) construíram gráficos da transmitância (T) em função da frequência do feixe incidente. A Figura 6b mostra os resultados para um conjunto de 4 camadas de materiais dielétricos ($N = 4$).

Figura 6 – Sistema óptico de N camadas dielétricas e resultados da simulação do sistema feita no software MATLAB.

- (a) Representação esquemática da propagação da luz no sistema de N camadas dielétricas. $E_+^{(i)}$ e $E_-^{(i)}$ são os campos elétricos da radiação se propagando para direita e para esquerda respectivamente. L_i é a i -ésima camada dielétrica.
- (b) Transmitância T em função da frequência normalizada para frequência central f_0 para as 10 configurações de 4 camadas possíveis de serem formadas pelos dois materiais dielétricos.



Fonte: Liu, Ukhtary e Saito (2017).

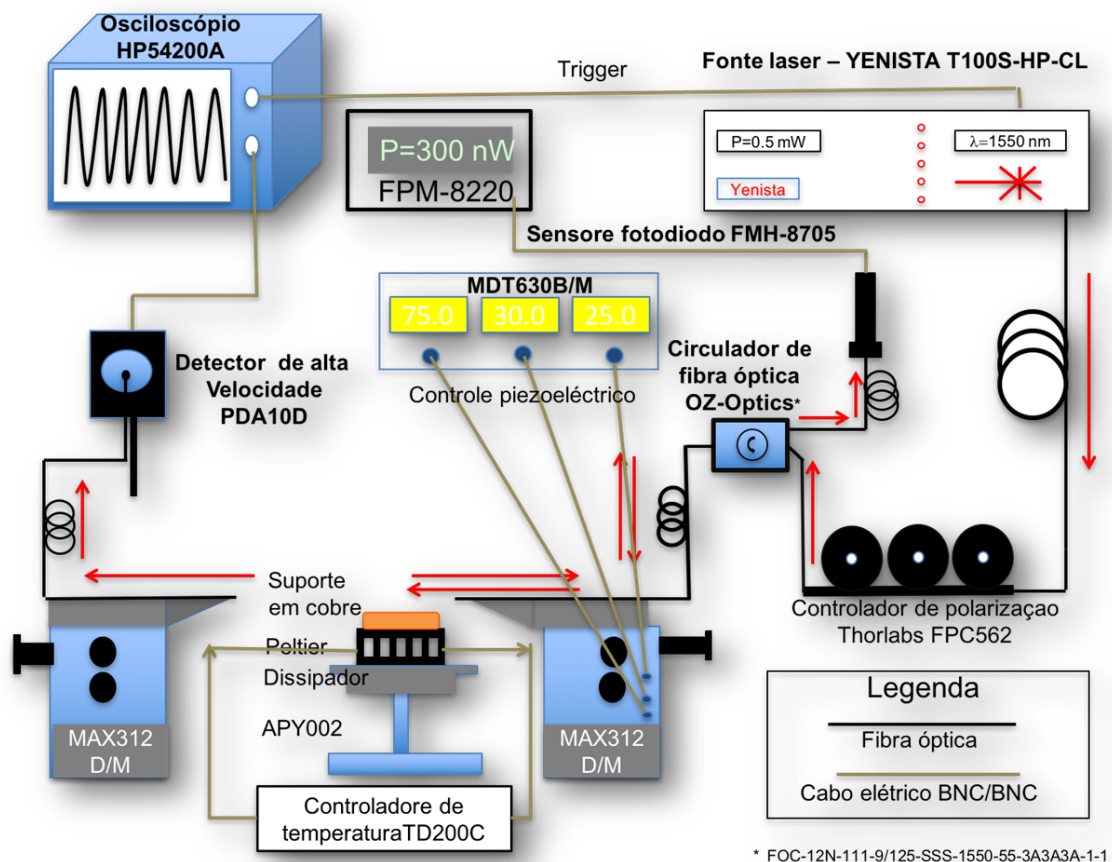
Os trabalhos citados nesta seção permitem inferir que a modelagem matemática do Interferômetro de Fabry-Pérot (IFP) feita pelo método matricial é a mais indicada para determinar as equações analíticas da transmitância para sistemas de múltiplas cavidades. Deste modo, o presente trabalho adotou tal método para estabelecer as relações teóricas e simular o comportamento de um IFP de três cavidades e compará-lo com o comportamento do experimento para

medição de perdas por espalhamento elaborado pelo Grupo de Óptica, Laser e Fotônica.

3 APARATO EXPERIMENTAL DA TÉCNICA PARA EXTRAIR AS PERDAS POR ESPALHAMENTO

A medição das perdas por espalhamento de materiais fotônicos passivos é realizada por meio do aparato experimental apresentado na Figura 7. Um feixe é transmitido por fibra óptica de uma fonte de luz coerente (laser), atravessa a guia de onda (amostra) e propaga-se por outra fibra até um detector que mede a potência luminosa recebida em μW .

Figura 7 – Aparato experimental da técnica para extrair as perdas por espalhamento.



Fonte: Grupo de Óptica, Laser e Fotônica.

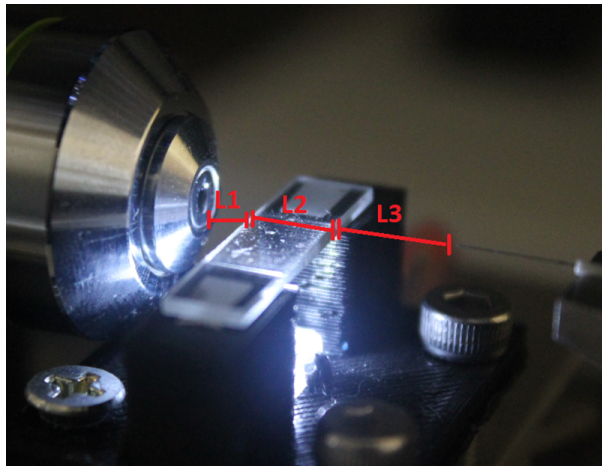
A luz, ao atravessar a guia de onda, sofre sucessivas reflexões internas devido às superfícies da guia, o que produz um padrão de interferência do tipo Fabry-Pérot, o qual é utilizado para determinar as perdas por espalhamento. Porém, é necessário fazer um estudo do setup experimental para evidenciar se há mais cavidades envolvidas no processo de medida e quais as suas contribuições. De fato, a análise do aparato experimental sugere que três cavidades ópticas devem ser contabilizadas:

- Cavidade entre a fibra e a guia de onda;

- Cavidade da própria guia de onda;
- Cavidade entre a guia e a fibra de saída do feixe.

Deste modo, para representar com precisão as varreduras realizadas pelo experimento, um Interferômetro de Fabry-Pérot de quatro interfaces reflexivas deve ser considerado. A Figura 8 mostra uma imagem do aparato experimental na região da guia de onda com as três cavidades sendo evidenciadas por $L1$, $L2$ e $L3$. As superfícies da guia de onda possuem refletividades de 0,3 - 0,5 e as superfícies das fibras ópticas, 0,05.

Figura 8 – Região da guia de onda (amostra) com as três cavidades evidenciadas.



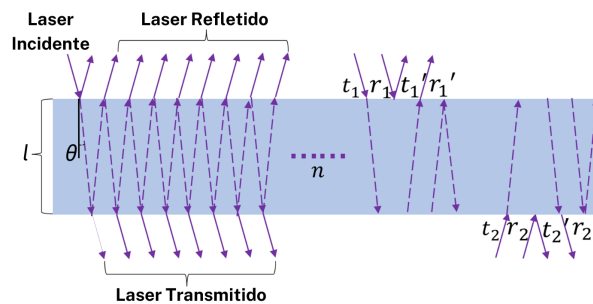
Fonte: Grupo de Óptica, Laser e Fotônica.

4 FUNDAMENTAÇÃO TEÓRICA

4.1 Interferômetro de Fabry-Perot (IFP)

Os físicos franceses Charles Fabry e Alfred Pérot construíram, no final do século XIX, um ressonador óptico para uso como interferômetro (FABRY; PÉROT, 1897). O sistema físico, chamado de Interferômetro de Fabry-Perot (IFP) em homenagem aos seus inventores, compreende duas interfaces refletoras planas e paralelas em que a luz é repetidamente refletida entre elas, experimentando poucas perdas e sofrendo interferências múltiplas. Deste modo, a luz é confinada, armazenada e transmitida em frequências de ressonância que são definidas pelas características do sistema (SALEH; TEICH, 2019). Um material transparente de superfícies planas pode constituir um IFP, tendo em vista que, internamente, a luz é refletida inúmeras vezes pelas superfícies. Tais materiais recebem o nome de etalon. Uma representação esquemática é apresentada na Figura 9.

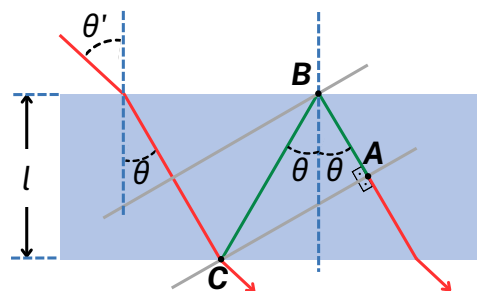
Figura 9 – Comportamento da propagação de um laser em um etalon.



Fonte: Du et al. (2023).

A modelagem matemática da interferência das ondas luminosas no etalon pode ser feita partindo-se da diferença entre o caminho óptico de feixes consecutivos (Δd), que pode ser deduzida da seguinte forma a partir da Figura 10:

Figura 10 – Representação do traçado de raios evidenciando a diferença de caminho óptico entre dois feixes consecutivos.



Fonte: Elaborado pelo próprio autor.

$$\begin{aligned}\cos(\theta) &= \frac{l}{\overline{BC}} \implies \overline{BC} = \frac{l}{\cos(\theta)} \\ \cos(2\theta) &= \frac{\overline{AB}}{\overline{BC}} \implies \overline{AB} = \overline{BC}\cos(2\theta) = \frac{l\cos(2\theta)}{\cos(\theta)} \\ \Delta d &= \overline{AB} + \overline{BC} = \frac{l\cos(2\theta)}{\cos(\theta)} + \frac{l}{\cos(\theta)} = \frac{l[\cos(2\theta) + 1]}{\cos(\theta)} = \frac{2l\cos^2(\theta)}{\cos(\theta)} = 2l\cos(\theta)\end{aligned}$$

Assim, a diferença de fase δ entre dois feixes consecutivos (a cada volta) pode ser expressa por:

$$\delta = K \cdot \Delta d = \frac{2\pi n}{\lambda} \cdot \Delta d = \frac{4\pi n l \cos(\theta)}{\lambda} \quad (1)$$

em que K é o número de onda, l é o comprimento da cavidade ou etalon, n o índice de refração do meio de propagação da luz e λ o comprimento de onda da radiação.

A Equação 1 representa a diferença de fase para um volta completa na cavidade (ida e volta). A partir dela, pode-se obter a diferença de fase ϕ para cada meia volta na cavidade (somente ida ou volta):

$$\phi = \frac{\delta}{2} = \frac{2\pi n l \cos(\theta)}{\lambda} \quad (2)$$

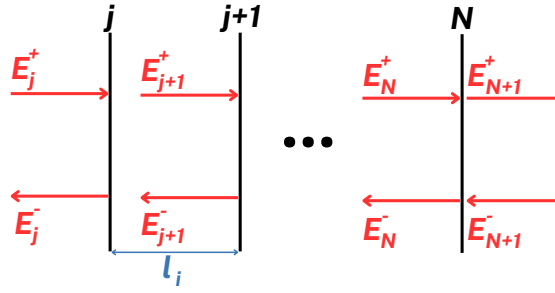
Para um feixe incidente perpendicular a superfície do etalon, ou seja, para $\theta = 0$, ϕ pode ser escrito como:

$$\phi = \frac{2\pi n l}{\lambda} \quad (3)$$

Seja um Interferômetro de Fabry Perot geral de N espelhos delgados planos e paralelos conforme representado na Figura 11. Na imagem, E_j^+ é a amplitude do vetor campo elétrico imediatamente no lado esquerdo da interface j para uma frente de onda que se propaga para a direita e E_j^- é a amplitude no lado esquerdo do espelho j para uma frente de onda viajando para a esquerda. A distância entre os espelhos j e $j + 1$ é igual a l_j que confere uma diferença de fase ϕ_j aos campos elétricos:

$$\phi_j = \frac{2\pi n_j l_j}{\lambda} \quad (4)$$

Para a análise, assume-se que não há perdas por absorção ou dispersão no interferômetro. Além disso, as seguintes relações devem ser consideradas:

Figura 11 – Representação esquemática de um IFP de N interfaces refletoras.


Fonte: Elaborado pelo próprio autor.

$$I = EE^* \quad (5)$$

$$r_j = r_j^* \quad (6)$$

$$t_j = t_j^* \quad (7)$$

$$R_j = r_j r_j^* = r_j^2 \quad (8)$$

$$T_j = t_j t_j^* = t_j^2 \quad (9)$$

$$R_j + T_j = 1 \quad (10)$$

em que I é a intensidade, E a amplitude do campo elétrico; R_j , r_j , T_j e t_j a refletividade, o coeficiente de reflexão, a transmissividade e o coeficiente de transmissão de cada interface j respectivamente.

A Figura 11 permite deduzir as seguintes relações para as amplitudes dos campos elétricos:

$$\begin{cases} E_{j+1}^+ = t_j E_j^+ e^{i\phi_j} + r_j E_{j+1}^- e^{i2\phi_j} \\ E_j^- = -r_j E_j^- + t_j E_{j+1}^- e^{i\phi_j} \end{cases} \quad (11)$$

A manipulação da primeira equação do sistema 11 permite isolar E_j^+ da seguinte maneira:

$$t_j E_j^+ = E_{j+1}^+ e^{-i\phi_j} - r_j E_{j+1}^- e^{i\phi_j} \rightarrow E_j^+ = \frac{1}{t_j} E_{j+1}^+ e^{-i\phi_j} - \frac{r_j}{t_j} E_{j+1}^- e^{i\phi_j} \quad (12)$$

Aplicando tal resultado na segunda equação do sistema 11, obtém-se uma nova relação:

$$E_j^- = -r_j \left[\frac{1}{t_j} E_{j+1}^+ e^{-i\phi_j} - \frac{r_j}{t_j} E_{j+1}^- e^{i\phi_j} \right] + t_j E_{j+1}^- e^{i\phi_j} \quad (13)$$

$$E_j^- = -\frac{r_j}{t_j} E_{j+1}^+ e^{-i\phi_j} + \frac{r_j^2}{t_j} E_{j+1}^- e^{i\phi_j} + t_j E_{j+1}^- e^{i\phi_j} \quad (14)$$

$$E_j^- = -\frac{r_j}{t_j} E_{j+1}^+ e^{-i\phi_j} + \frac{r_j^2 + t_j^2}{t_j} E_{j+1}^- e^{i\phi_j} \quad (15)$$

$$E_j^- = -\frac{r_j}{t_j} E_{j+1}^+ e^{-i\phi_j} + \frac{1}{t_j} E_{j+1}^- e^{i\phi_j} \quad (16)$$

Logo, um novo sistema pode ser escrito:

$$\begin{cases} E_j^+ &= \frac{1}{t_j} \left(E_{j+1}^+ e^{-i\phi_j} - r_j E_{j+1}^- e^{i\phi_j} \right) \\ E_j^- &= \frac{1}{t_j} \left(-r_j E_{j+1}^+ e^{-i\phi_j} + E_{j+1}^- e^{i\phi_j} \right) \end{cases} \quad (17)$$

O sistema pode ser reescrito na forma matricial como:

$$\begin{pmatrix} E_j^+ \\ E_j^- \end{pmatrix} = \frac{1}{t_j} \begin{pmatrix} e^{-i\phi_j} & -r_j e^{i\phi_j} \\ -r_j e^{-i\phi_j} & e^{i\phi_j} \end{pmatrix} \begin{pmatrix} E_{j+1}^+ \\ E_{j+1}^- \end{pmatrix}$$

Assim, a matriz de transferência (ou matriz ABCD) de uma cavidade é:

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} e^{-i\phi_j} & -r_j e^{i\phi_j} \\ -r_j e^{-i\phi_j} & e^{i\phi_j} \end{pmatrix}$$

Uma vez determinada a matriz ABCD para uma cavidade, que é o elemento básico de um Interferômetro de Fabry-Perot (IFP), pode-se estabelecer a matriz de transferência de um IFP com um número qualquer de cavidades. Para isso, aplica-se a matriz de transformação para cada cavidade presente no IFP, ou seja, a matriz ABCD é multiplicada $N - 1$ vezes (quantidade de cavidades).

No caso geral de um IFP de N superfícies de coeficiente de reflexão r_j e transmissão t_j e comprimentos das cavidades iguais a l_j , tem-se a seguinte relação:

$$\begin{aligned} \begin{pmatrix} E_j^+ \\ E_j^- \end{pmatrix} &= \frac{1}{t_1} \begin{pmatrix} e^{-i\phi_1} & -r_1 e^{i\phi_1} \\ -r_1 e^{-i\phi_1} & e^{i\phi_1} \end{pmatrix} \frac{1}{t_2} \begin{pmatrix} e^{-i\phi_2} & -r_2 e^{i\phi_2} \\ -r_2 e^{-i\phi_2} & e^{i\phi_2} \end{pmatrix} \frac{1}{t_3} \begin{pmatrix} e^{-i\phi_3} & -r_3 e^{i\phi_3} \\ -r_3 e^{-i\phi_3} & e^{i\phi_3} \end{pmatrix} \cdots \\ &\cdots \frac{1}{t_{N-1}} \begin{pmatrix} e^{-i\phi_{N-1}} & -r_{N-1} e^{i\phi_{N-1}} \\ -r_{N-1} e^{-i\phi_{N-1}} & e^{i\phi_{N-1}} \end{pmatrix} \begin{pmatrix} E_N^+ \\ E_N^- \end{pmatrix} \\ \begin{pmatrix} E_j^+ \\ E_j^- \end{pmatrix} &= \frac{1}{t_1 t_2 t_3 \cdots t_{N-1}} \prod_{j=1}^{N-1} \begin{pmatrix} e^{-i\phi_j} & -r_j e^{i\phi_j} \\ -r_j e^{-i\phi_j} & e^{i\phi_j} \end{pmatrix} \begin{pmatrix} E_N^+ \\ E_N^- \end{pmatrix} \end{aligned} \quad (18)$$

Desta forma, para o caso geral, a matriz ABCD do IFP de múltiplas cavidades é:

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \prod_{j=1}^{N-1} \begin{pmatrix} e^{-i\phi_j} & -r_j e^{i\phi_j} \\ -r_j e^{-i\phi_j} & e^{i\phi_j} \end{pmatrix} \quad (19)$$

Pela Equação 18 é possível determinar o coeficiente de transmissão equivalente t para o interferômetro:

$$t = \frac{E_{N+1}^+}{E_1^+} = \frac{t_N E_N^+}{E_1^+} = \frac{t_N E_N^+}{\frac{A E_N^+ + B E_N^+}{t_1 t_2 t_3 \cdots t_{N-1}}} = \frac{t_1 t_2 t_3 \cdots t_N E_N^+}{A E_N^+ + B(-r_N E_N^+)} = \frac{t_1 t_2 t_3 \cdots t_N}{A - r_N B} \quad (20)$$

4.2 Interferômetro de Fabry-Perot de 3 cavidades

A técnica para medição das perdas por espelhamento de materiais passivos usados em fotônica desenvolvida no OLAF constitui um IFP de 3 cavidades, ou seja, $N = 4$. A fim de se obter o coeficiente de transmissão t , é necessário determinar os coeficiente A e B da matriz ABCD do sistema. Pela Equação 19, os coeficientes podem ser encontrados da seguinte maneira:

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} e^{-i\phi_1} & -r_1 e^{i\phi_1} \\ -r_1 e^{-i\phi_1} & e^{i\phi_1} \end{pmatrix} \begin{pmatrix} e^{-i\phi_2} & -r_2 e^{i\phi_2} \\ -r_2 e^{-i\phi_2} & e^{i\phi_2} \end{pmatrix} \begin{pmatrix} e^{-i\phi_3} & -r_3 e^{i\phi_3} \\ -r_3 e^{-i\phi_3} & e^{i\phi_3} \end{pmatrix} \quad (21)$$

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} e^{-i(\phi_1+\phi_2+\phi_3)} + r_1 r_2 e^{i(\phi_1-\phi_2-\phi_3)} + r_2 r_3 e^{i(-\phi_1+\phi_2-\phi_3)} + r_1 r_3 e^{i(\phi_1+\phi_2-\phi_3)} \\ -r_1 e^{-i(\phi_1+\phi_2+\phi_3)} - r_2 e^{i(\phi_1-\phi_2-\phi_3)} - r_1 r_2 r_3 e^{i(-\phi_1+\phi_2-\phi_3)} - r_3 e^{i(\phi_1+\phi_2-\phi_3)} \\ -r_3 e^{-i(\phi_1+\phi_2-\phi_3)} - r_1 r_2 r_3 e^{i(\phi_1-\phi_2+\phi_3)} - r_2 e^{i(-\phi_1+\phi_2+\phi_3)} - r_1 e^{i(\phi_1+\phi_2+\phi_3)} \\ r_1 r_3 e^{-i(\phi_1+\phi_2-\phi_3)} + r_2 r_3 e^{i(\phi_1-\phi_2+\phi_3)} r_1 r_2 e^{i(-\phi_1+\phi_2+\phi_3)} + e^{i(\phi_1+\phi_2+\phi_3)} \end{pmatrix}$$

ou seja,

$$A = e^{-i(\phi_1+\phi_2+\phi_3)} + r_1 r_2 e^{-i(-\phi_1+\phi_2+\phi_3)} + r_2 r_3 e^{-i(\phi_1-\phi_2+\phi_3)} + r_1 r_3 e^{-i(-\phi_1-\phi_2+\phi_3)} \quad (22)$$

$$B = -r_3 e^{-i(\phi_1+\phi_2-\phi_3)} - r_1 r_2 r_3 e^{i(\phi_1-\phi_2+\phi_3)} - r_2 e^{i(-\phi_1+\phi_2+\phi_3)} - r_1 e^{i(\phi_1+\phi_2+\phi_3)} \quad (23)$$

Nessa caso de $N = 4$, a relação para t é definida utilizando a Equação 20:

$$t = \frac{t_1 t_2 t_3 t_4}{A - r_4 B} \quad (24)$$

O que realmente é verificado em laboratório é a intensidade relativa transmitida T do sistema, que pode ser obtida da seguinte forma:

$$T = tt^* = \left(\frac{t_1 t_2 t_3 t_4}{A - r_4 B} \right) \left(\frac{t_1^* t_2^* t_3^* t_4^*}{A^* - r_4^* B^*} \right) = \frac{(t_1 t_2 t_3 t_4)^2}{D_4} \quad (25)$$

sendo que

$$D_4 = (A - r_4 B)(A^* - r_4^* B^*) = AA^* - r_4(AB^* + A^*B) + r_4^2 BB^* \quad (26)$$

Para determinar D_4 , calcula-se cada termo da Equação 26:

$$AA^* = \left[e^{-i(\phi_1 + \phi_2 + \phi_3)} + r_1 r_2 e^{-i(-\phi_1 + \phi_2 + \phi_3)} + r_2 r_3 e^{-i(\phi_1 - \phi_2 + \phi_3)} + r_1 r_3 e^{-i(-\phi_1 - \phi_2 + \phi_3)} \right] \times \\ \left[e^{i(\phi_1 + \phi_2 + \phi_3)} + r_1 r_2 e^{i(-\phi_1 + \phi_2 + \phi_3)} + r_2 r_3 e^{i(\phi_1 - \phi_2 + \phi_3)} + r_1 r_3 e^{i(-\phi_1 - \phi_2 + \phi_3)} \right]$$

$$AA^* = 1 + r_1 r_2 e^{-i2\phi_1} + r_1 r_3 e^{-i2(\phi_1 + \phi_2)} + r_2 r_3 e^{-2\phi_2} + r_1^2 r_2^2 + r_1^2 r_2 r_3 e^{-i2\phi_2} \\ + r_1 r_2^2 r_3 e^{i2(\phi_1 - \phi_2)} + r_1 r_2 e^{i2\phi_1} + r_1 r_2^2 r_3 e^{i2(\phi_2 - \phi_1)} + r_1 r_2 r_3^2 e^{-i2\phi_1} + r_2^2 r_3^2 \\ + r_2 r_3 e^{i2\phi_2} + r_1^2 r_2 r_3 e^{i2\phi_2} + r_1^2 r_3^2 + r_1 r_2 r_3^2 e^{i2\phi_1} + r_1 r_3 e^{i2(\phi_1 + \phi_2)}$$

$$AA^* = 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + (r_1 r_2 + r_1 r_2 r_3^2) e^{-i2\phi_1} + (r_1 r_2 + r_1 r_2 r_3^2) e^{i2\phi_1} \\ + (r_2 r_3 + r_1^2 r_2 r_3) e^{-i2\phi_2} + (r_2 r_3 + r_1^2 r_2 r_3) e^{i2\phi_2} + r_1 r_3 \left[e^{-i2(\phi_1 + \phi_2)} + e^{-2(\phi_1 + \phi_2)} \right] \\ + r_1 r_2^2 r_3 \left[e^{i2(\phi_1 - \phi_2)} + e^{-i2(\phi_1 - \phi_2)} \right]$$

$$AA^* = 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + r_1 r_2 (r_3^2 + 1) \left[e^{i2\phi_1} + e^{-i2\phi_1} \right] \\ + r_2 r_3 (r_1^2 + 1) \left[e^{i2\phi_2} + e^{-i2\phi_2} \right] + r_1 r_3 \left[e^{i2(\phi_1 + \phi_2)} + e^{-i2(\phi_1 + \phi_2)} \right] \\ + r_1 r_2^2 r_3 \left[e^{i2(\phi_1 - \phi_2)} + e^{i2(\phi_1 - \phi_2)} \right]$$

$$BB^* = \left[-r_3 e^{-i(\phi_1 + \phi_2 - \phi_3)} - r_1 r_2 r_3 e^{i(\phi_1 - \phi_2 + \phi_3)} - r_2 e^{i(-\phi_1 + \phi_2 + \phi_3)} - r_1 e^{i(\phi_1 + \phi_2 + \phi_3)} \right] \times \\ \left[-r_3 e^{i(\phi_1 + \phi_2 - \phi_3)} - r_1 r_2 r_3 e^{-i(\phi_1 - \phi_2 + \phi_3)} - r_2 e^{-i(-\phi_1 + \phi_2 + \phi_3)} - r_1 e^{-i(\phi_1 + \phi_2 + \phi_3)} \right]$$

$$BB^* = r_3^2 + r_1 r_2 r_3^2 e^{-i2\phi_1} + r_2 r_3 e^{-i2\phi_2} + r_1 r_3 e^{-i2(\phi_1 + \phi_2)} + r_1 r_2 r_3^2 e^{i2\phi_1} + r_1^2 r_2^2 r_3^2 \\ + r_1 r_2^2 r_3 e^{i2(\phi_1 - \phi_2)} + r_1^2 r_2 r_3 e^{-i2\phi_2} + r_2 r_3 e^{i2\phi_2} + r_1 r_2^2 r_3 e^{-i2(\phi_1 - \phi_2)} \\ + r_2^2 + r_1 r_2 e^{-2\phi_1} + r_1 r_3 e^{i2(\phi_1 + \phi_2)} + r_1^2 r_2 r_3 e^{i2\phi_2} + r_1 r_2 e^{i2\phi_1} + r_1^2$$

$$BB^* = r_1^2 + r_2^2 + r_3^2 + (r_1 r_2 r_3^2 + r_1 r_2) e^{-i2\phi_1} + (r_1 r_2 r_3^2 + r_1 r_2) e^{i2\phi_2} + (r_2 r_3 + r_1^2 r_2 r_3) e^{i2\phi_2} \\ + r_1 r_3 \left[e^{i2(\phi_1 + \phi_2)} + e^{-i2(\phi_1 + \phi_2)} \right] + r_1 r_2^2 r_3 \left[e^{i2(\phi_1 - \phi_2)} + e^{-i2(\phi_1 - \phi_2)} \right] + (r_1 r_2 r_3)^2$$

$$BB^* = r_1^2 + r_2^2 + r_3^2 + (r_1 r_2 r_3)^2 + r_1 r_2 (r_3^2 + 1) \left[e^{i2\phi_1} + e^{-i2\phi_1} \right] + r_2 r_3 (r_1^2 + 1) \left[e^{i2\phi_2} + e^{-i2\phi_2} \right] \\ + r_1 r_3 \left[e^{i2(\phi_1 + \phi_2)} + e^{-i2(\phi_1 + \phi_2)} \right] + r_1 r_2^2 r_3 \left[e^{i2(\phi_1 - \phi_2)} + e^{-i2(\phi_1 - \phi_2)} \right]$$

$$\begin{aligned}
AA^* + r_4^2 BB^* &= 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + r_1 r_2 (r_3^2 + 1) \left[e^{i2\phi_1} + e^{-i2\phi_1} \right] \\
&\quad + r_2 r_3 (r_1^2 + 1) \left[e^{i2\phi_2} + e^{-i2\phi_2} \right] + r_1 r_3 \left[e^{i2(\phi_1 + \phi_2)} + e^{-i2(\phi_1 + \phi_2)} \right] \\
&\quad + r_1 r_2^2 r_3 \left[e^{i2(\phi_1 - \phi_2)} + e^{i2(\phi_1 - \phi_2)} \right] + (r_1 r_4)^2 + (r_2 r_4)^2 + (r_3 r_4)^2 + (r_1 r_2 r_3 r_4)^2 \\
&\quad + r_1 r_2 r_4^2 (r_3^2 + 1) \left[e^{i2\phi_1} + e^{-i2\phi_1} \right] + r_2 r_3 r_4^2 (r_1^2 + 1) \left[e^{i2\phi_2} + e^{-i2\phi_2} \right] \\
&\quad + r_1 r_3 r_4^2 \left[e^{i2(\phi_1 + \phi_2)} + e^{-i2(\phi_1 + \phi_2)} \right] + r_1 r_2^2 r_3 r_4^2 \left[e^{i2(\phi_1 - \phi_2)} + e^{-i2(\phi_1 - \phi_2)} \right] \\
AA^* + r_4^2 BB^* &= 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + (r_1 r_4)^2 + (r_2 r_4)^2 + (r_3 r_4)^2 + (r_1 r_2 r_3 r_4)^2 \\
&\quad + r_1 r_2 (r_3^2 + 1) (r_4^2 + 1) \left[e^{i2\phi_1} + e^{-i2\phi_1} \right] \\
&\quad + r_2 r_3 (r_1^2 + 1) (r_4^2 + 1) \left[e^{i2\phi_2} + e^{-i2\phi_2} \right] \\
&\quad + r_1 r_3 (r_4^2 + 1) \left[e^{i2(\phi_1 + \phi_2)} + e^{-i2(\phi_1 + \phi_2)} \right] \\
&\quad + r_1 r_2^2 r_3 (r_4^2 + 1) \left[e^{i2(\phi_1 - \phi_2)} + e^{i2(\phi_1 - \phi_2)} \right]
\end{aligned}$$

Considerando a relação trigonométrica: $2\cos(x) = e^{ix} + e^{-ix}$, a expressão acima torna-se:

$$\begin{aligned}
AA^* + r_4^2 BB^* &= 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + (r_1 r_4)^2 + (r_2 r_4)^2 + (r_3 r_4)^2 + (r_1 r_2 r_3 r_4)^2 \\
&\quad + 2r_1 r_2 (r_3^2 + 1) (r_4^2 + 1) [\cos(2\phi_1)] + 2r_2 r_3 (r_1^2 + 1) (r_4^2 + 1) [\cos(2\phi_2)] \\
&\quad + 2r_1 r_3 (r_4^2 + 1) [\cos(2(\phi_1 + \phi_2))] + 2r_1 r_2^2 r_3 (r_4^2 + 1) [\cos(2(\phi_1 - \phi_2))]
\end{aligned}$$

Agora

$$\begin{aligned}
A^* B &= \left[e^{i(\phi_1 + \phi_2 + \phi_3)} + r_1 r_2 e^{i(-\phi_1 + \phi_2 + \phi_3)} + r_2 r_3 e^{i(\phi_1 - \phi_2 + \phi_3)} + r_1 r_3 e^{i(-\phi_1 - \phi_2 + \phi_3)} \right] \cdot \\
&\quad \left[-r_3 e^{-i(\phi_1 + \phi_2 - \phi_3)} - r_1 r_2 r_3 e^{i(\phi_1 - \phi_2 + \phi_3)} - r_2 e^{i(-\phi_1 + \phi_2 + \phi_3)} - r_1 e^{i(\phi_1 + \phi_2 + \phi_3)} \right] \\
A^* B &= -r_3 e^{i2\phi_3} - r_1 r_2 r_3 e^{i2(\phi_1 + \phi_3)} - r_2 e^{i2(\phi_2 + \phi_3)} - r_1 e^{i2(\phi_1 + \phi_2 + \phi_3)} - r_1 r_2 r_3 e^{i2(-\phi_1 + \phi_3)} \\
&\quad - r_1^2 r_2^2 r_3 e^{i2\phi_3} - r_1 r_2^2 e^{i2(-\phi_1 + \phi_2 + \phi_3)} - r_1^2 r_2 e^{i2(\phi_2 + \phi_3)} - r_2 r_3^2 e^{i2(-\phi_2 + \phi_3)} \\
&\quad - r_1 r_2^2 r_3^2 e^{i2(\phi_1 - \phi_2 + \phi_3)} - r_2^2 r_3 e^{i2\phi_3} - r_1 r_2 r_3 e^{i2(\phi_1 + \phi_3)} - r_1 r_3^2 e^{i2(-\phi_1 - \phi_2 + \phi_3)} \\
&\quad - r_1 r_2 r_3^2 e^{i2(-\phi_2 + \phi_3)} - r_1 r_2 r_3 e^{i2(-\phi_1 + \phi_3)} - r_1^2 r_3 e^{i2\phi_3} \\
A^* B &= -(r_3 + r_1^2 r_2^2 r_3 + r_2^2 r_3 + r_1^2 r_3) e^{i2\phi_3} - 2r_1 r_2 r_3 e^{i2(\phi_1 + \phi_3)} - (r_2 + r_1^2 r_2) e^{i2(\phi_2 + \phi_3)} \\
&\quad - r_1 e^{i2(\phi_1 + \phi_2 + \phi_3)} - 2r_1 r_2 r_3 e^{i2(-\phi_1 + \phi_3)} - r_1 r_2^2 e^{i2(-\phi_1 + \phi_2 + \phi_3)} \\
&\quad - (r_2 r_3^2 + r_1^2 r_2 r_3^2) e^{i2(-\phi_2 + \phi_3)} - r_1 r_2^2 r_3 e^{i2(\phi_1 - \phi_2 + \phi_3)} - r_1 r_3^2 e^{i2(-\phi_1 - \phi_2 + \phi_3)}
\end{aligned}$$

$$\begin{aligned}
AB^* = (A^*B)^* &= -(r_3 + r_1^2 r_2^2 r_3 + r_2^2 r_3 + r_1^2 r_3) e^{-i2\phi_3} - 2r_1 r_2 r_3 e^{-i2(\phi_1 + \phi_3)} \\
&\quad - (r_2 + r_1^2 r_2) e^{-i2(\phi_2 + \phi_3)} - r_1 e^{-i2(\phi_1 + \phi_2 + \phi_3)} - 2r_1 r_2 r_3 e^{-i2(-\phi_1 + \phi_3)} \\
&\quad - r_1 r_2^2 e^{-i2(-\phi_1 + \phi_2 + \phi_3)} - (r_2 r_3^2 + r_1^2 r_2 r_3^2) e^{-i2(-\phi_2 + \phi_3)} \\
&\quad - r_1 r_2^2 r_3^2 e^{-i2(\phi_1 - \phi_2 + \phi_3)} - r_1 r_3^2 e^{-i2(-\phi_1 - \phi_2 + \phi_3)}
\end{aligned}$$

$$\begin{aligned}
A^*B + AB^* &= -r_3(1 + r_1^2 r_2^2 + r_1^2) \left[e^{i2\phi_3} + e^{-i2\phi_3} \right] - 2r_1 r_2 r_3 \left[i2(\phi_1 + \phi_3) + e^{-i2(\phi_1 + \phi_3)} \right] \\
&\quad - r_2(r_1^2 + 1) \left[e^{i2(\phi_2 + \phi_3)} + e^{-i2(\phi_2 + \phi_3)} \right] - r_1 \left[e^{i2(\phi_1 + \phi_2 + \phi_3)} + e^{-i2(\phi_1 + \phi_2 + \phi_3)} \right] \\
&\quad - 2r_1 r_2 r_3 \left[e^{i2(-\phi_1 + \phi_3)} + e^{-i2(-\phi_1 + \phi_3)} \right] \\
&\quad - r_1 r_2^2 \left[e^{i2(-\phi_1 + \phi_2 + \phi_3)} + e^{-i2(-\phi_1 + \phi_2 + \phi_3)} \right] \\
&\quad - r_2 r_3^2 (r_1^2 + 1) \left[e^{i2(-\phi_2 + \phi_3)} + e^{-i2(-\phi_2 + \phi_3)} \right] \\
&\quad - r_1 r_2^2 r_3^2 \left[e^{i2(\phi_1 - \phi_2 + \phi_3)} + e^{i2(\phi_1 - \phi_2 + \phi_3)} \right] \\
&\quad - r_1 r_3^2 \left[e^{i2(-\phi_1 - \phi_2 + \phi_3)} + e^{-i2(-\phi_1 - \phi_2 + \phi_3)} \right]
\end{aligned}$$

Usando novamente a relação $2\cos(x) = e^{ix} + e^{-ix}$, tem-se:

$$\begin{aligned}
A^*B + AB^* &= -2r_3(r_1^2 + 1)(r_2^2 + 1)\cos(2\phi_3) - 4r_1 r_2 r_3 \cos(2(\phi_1 + \phi_3)) \\
&\quad - 2r_2(r_1^2 + 1)\cos(2(\phi_2 + \phi_3)) - 2r_1 \cos(2(\phi_1 + \phi_2 + \phi_3)) \\
&\quad - 4r_1 r_2 r_3 \cos(2(-\phi_1 + \phi_3)) - 2r_1 r_2^2 \cos(2(-\phi_1 + \phi_2 + \phi_3)) \\
&\quad - 2r_2 r_3^2 (r_1^2 + 1)\cos(2(-\phi_2 + \phi_3)) - 2r_1 r_2^2 r_3^2 \cos(2(\phi_1 - \phi_2 + \phi_3)) \\
&\quad - 2r_1 r_3^2 \cos(2(-\phi_1 - \phi_2 + \phi_3))
\end{aligned}$$

Finalmente

$$\begin{aligned}
D_4 &= AA^* + r_4^2 BB^* - r_4(A^*B + AB^*) \\
D_4 &= 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + (r_1 r_4)^2 + (r_2 r_4)^2 + (r_3 r_4)^2 + (r_1 r_2 r_3 r_4)^2 \\
&\quad + 2r_1 r_2 (r_3^2 + 1)(r_4^2 + 1)\cos(2\phi_1) + 2r_2 r_3 (r_1^2 + 1)(r_4^2 + 1)\cos(2\phi_2) \\
&\quad + 2r_1 r_3 (r_4^2 + 1)\cos(2(\phi_1 + \phi_2)) + 2r_1 r_2^2 r_3 (r_4^2 + 1)\cos(2(\phi_1 - \phi_2)) \\
&\quad + 2r_3 r_4 (r_1^2 + 1)(r_2^2 + 1)\cos(2\phi_3) + 4r_1 r_2 r_3 r_4 \cos(2(\phi_1 + \phi_3)) \\
&\quad + 2r_2 r_4 (r_1^2 + 1)\cos(2(\phi_2 + \phi_3)) + 2r_1 r_4 \cos(2(\phi_1 + \phi_2 + \phi_3)) \\
&\quad + 4r_1 r_2 r_3 r_4 \cos(2(-\phi_1 + \phi_3)) + 2r_1 r_2^2 r_4 \cos(2(-\phi_1 + \phi_2 + \phi_3)) \\
&\quad + 2r_2 r_3^2 r_4 (r_1^2 + 1)\cos(2(-\phi_2 + \phi_3)) + 2r_1 r_2^2 r_3^2 r_4 \cos(2(\phi_1 - \phi_2 + \phi_3)) \\
&\quad + 2r_1 r_3^2 r_4 \cos(2(-\phi_1 - \phi_2 + \phi_3))
\end{aligned}$$

A função cosseno é uma função par, ou seja, $\cos(x) = \cos(-x)$. Tal propriedade implica que: $\cos(2(-\phi_1 + \phi_3)) = \cos(2(\phi_1 - \phi_3))$, $\cos(2(-\phi_1 + \phi_2 + \phi_3)) = \cos(2(\phi_1 - \phi_2 - \phi_3))$, $\cos(2(-\phi_2 + \phi_3)) = \cos(2(\phi_2 - \phi_3))$ e $\cos(2(-\phi_1 - \phi_2 + \phi_3)) = \cos(2(\phi_1 + \phi_2 - \phi_3))$. Assim D_4 pode ser reescrito como:

$$\begin{aligned}
D_4 = & 1 + (r_1 r_2)^2 + (r_2 r_3)^2 + (r_1 r_3)^2 + (r_1 r_4)^2 + (r_2 r_4)^2 + (r_3 r_4)^2 + (r_1 r_2 r_3 r_4)^2 \\
& + 2r_1 r_2 (r_3^2 + 1)(r_4^2 + 1) \cos(2\phi_1) + 2r_2 r_3 (r_1^2 + 1)(r_4^2 + 1) \cos(2\phi_2) \\
& + 2r_1 r_3 (r_4^2 + 1) \cos(2(\phi_1 + \phi_2)) + 2r_1 r_2^2 r_3 (r_4^2 + 1) \cos(2(\phi_1 - \phi_2)) \\
& + 2r_3 r_4 (r_1^2 + 1)(r_2^2 + 1) \cos(2\phi_3) + 4r_1 r_2 r_3 r_4 \cos(2(\phi_1 + \phi_3)) \\
& + 2r_2 r_4 (r_1^2 + 1) \cos(2(\phi_2 + \phi_3)) + 2r_1 r_4 \cos(2(\phi_1 + \phi_2 + \phi_3)) \\
& + 4r_1 r_2 r_3 r_4 \cos(2(\phi_1 - \phi_3)) + 2r_1 r_2^2 r_4 \cos(2(\phi_1 - \phi_2 - \phi_3)) \\
& + 2r_2 r_3^2 r_4 (r_1^2 + 1) \cos(2(\phi_2 - \phi_3)) + 2r_1 r_2^2 r_3^2 r_4 \cos(2(\phi_1 - \phi_2 + \phi_3)) \\
& + 2r_1 r_3^2 r_4 \cos(2(\phi_1 + \phi_2 - \phi_3))
\end{aligned}$$

Por fim, utilizando as relações $R_j = r_j^2$, $T_j = t_j^2$ e $R + T = 1$, tem-se que

$$T = \frac{(t_1 t_2 t_3 t_4)^2}{D_4} = \frac{T_1 T_2 T_3 T_4}{D_4} = \frac{(1 - R_1)(1 - R_2)(1 - R_3)(1 - R_4)}{D_4} \quad (27)$$

sendo que

$$\begin{aligned}
D_4 = & 1 + R_1 R_2 + R_2 R_3 + R_1 R_3 + R_1 R_4 + R_2 R_4 + R_3 R_4 + R_1 R_2 R_3 R_4 \\
& + 2\sqrt{R_1 R_2} (R_3 + 1)(R_4 + 1) \cos(2\phi_1) + 2\sqrt{R_2 R_3} (R_1 + 1)(R_4 + 1) \cos(2\phi_2) \\
& + 2\sqrt{R_1 R_3} (R_4 + 1) \cos(2(\phi_1 + \phi_2)) + 2R_2 \sqrt{R_1 R_3} (R_4 + 1) \cos(2(\phi_1 - \phi_2)) \\
& + 2\sqrt{R_3 R_4} (R_1 + 1)(R_2 + 1) \cos(2\phi_3) + 4\sqrt{R_1 R_2 R_3 R_4} \cos(2(\phi_1 + \phi_3)) \\
& + 2\sqrt{R_2 R_4} (R_1 + 1) \cos(2(\phi_2 + \phi_3)) + 2\sqrt{R_1 R_4} \cos(2(\phi_1 + \phi_2 + \phi_3)) \\
& + 4\sqrt{R_1 R_2 R_3 R_4} \cos(2(\phi_1 - \phi_3)) + 2R_2 \sqrt{R_1 R_4} \cos(2(\phi_1 - \phi_2 - \phi_3)) \\
& + 2R_3 \sqrt{R_2 R_4} (R_1 + 1) \cos(2(\phi_2 - \phi_3)) + 2R_2 R_3 \sqrt{R_1 R_4} \cos(2(\phi_1 - \phi_2 + \phi_3)) \\
& + 2R_3 \sqrt{R_1 R_4} \cos(2(\phi_1 + \phi_2 - \phi_3))
\end{aligned}$$

A partir da Equação 27 e recordando a Equação 4, os parâmetros a serem estabelecidos para gerar o espectro de transmissão são:

- As refletividades de cada interface R_1 , R_2 , R_3 e R_4 ;
- Os comprimentos das cavidades l_1 , l_2 e l_3 ;
- Os índices de refração dos meios que compõem cada cavidade n_1 , n_2 e n_3 .

O espectro de transmissão do IFP descrito pela Equação 27 é uma composição de funções harmônicas de frequências fundamentais f dadas por

$$f = \frac{n_j l_j}{\pi}.$$

A transformada de Fourier (TF) do espectro T em função do número de onda k permite a extração dos valores de tais frequências, uma vez que o gráfico da TF apresenta picos nessas posições (HOFSTETTER; THORNTON, 1997). Deste modo, a partir da curva T , pode-se determinar o comprimento óptico de uma cavidade, cujo valor é $n_j l_j$. Sabendo o índice de refração da cavidade, obtém-se o seu comprimento real.

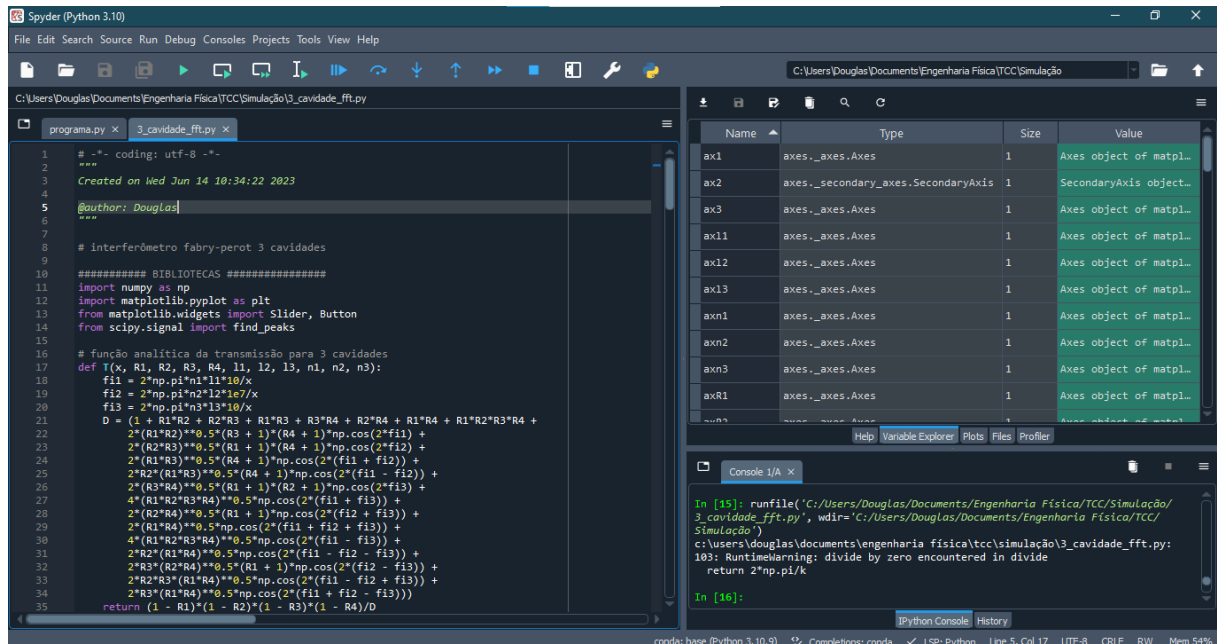
5 FERRAMENTAS E MÉTODOS

5.1 Ferramentas

Para a execução deste trabalho foi utilizada a linguagem de programação **Python**. Ela consiste em uma linguagem de alto nível, interpretada, orientada a objetos e de propósito geral. O Python destaca-se por ter uma sintaxe clara e legível, que favorece a escrita de um código conciso e de fácil compreensão (MENEZES, 2019). Além disso, é um software de código aberto, o que significa que pode ser utilizado gratuitamente. Isso é possível graças ao esforço da Python Foundation¹ e à contribuição de diversos colaboradores.

Como interpretador da linguagem, foi utilizado o software **Spyder**². Segundo o site dos desenvolvedores, o Spyder é um ambiente de desenvolvimento integrado (IDE - *integrated development environment*) especialmente projetado para programação em Python e análise de dados científicos. Ele fornece uma interface gráfica que reúne várias ferramentas e recursos úteis para facilitar o desenvolvimento, depuração e análise de código Python, especialmente em contextos científicos e de análise de dados. O layout do software Spyder é exibido na Figura 12.

Figura 12 – Layout do IDE Spyder.



Fonte: Elaborado pelo próprio autor.

¹ <<https://www.python.org/psf/mission/>>

² <<https://www.spyder-ide.org/>>

5.2 Métodos

Como citado no Capítulo 1, este trabalho tem o objetivo de auxiliar a análise dos dados da técnica de medição de perdas por espalhamento em materiais passivos aplicados à fotônica e identificar os parâmetros do Interferômetro de Fabry-Pérot (IFP) estabelecido pelo sistema experimental. A fim de desenvolver um software para cumprir tal finalidade, foi realizada a seguinte sequência de atividades:

- Revisão bibliográfica dos trabalhos que estabeleçam a equação teórica da transmitância T para um IFP de múltiplas cavidade ópticas para o entendimento do método matricial utilizado para os cálculos;
- Dedução da equação analítica (Equação 27) para um IFP de três cavidades, conforme apresentado na seção 4.2;
- Desenvolvimento de um software para simulação da transmitância T do IFP em função do comprimento de onda λ da luz incidente com a possibilidade do ajuste dos parâmetros R_1 , R_2 , R_3 , R_4 , l_1 , l_2 , l_3 , n_1 , n_2 e n_3 .
- Construção de um software com o intuito de analisar os dados experimentais e fornecer os valores dos parâmetros do IFP do experimento, comparando os resultados com a curva teórica. O programa contará com uma interface gráfica para a utilização de usuários por parte do Laboratório de Óptica, Laser e Fotônica.
- Análise de alguns arquivos de dados do experimento de medição de perdas por espalhamento para validação do software.

6 RESULTADOS E DISCUSSÃO

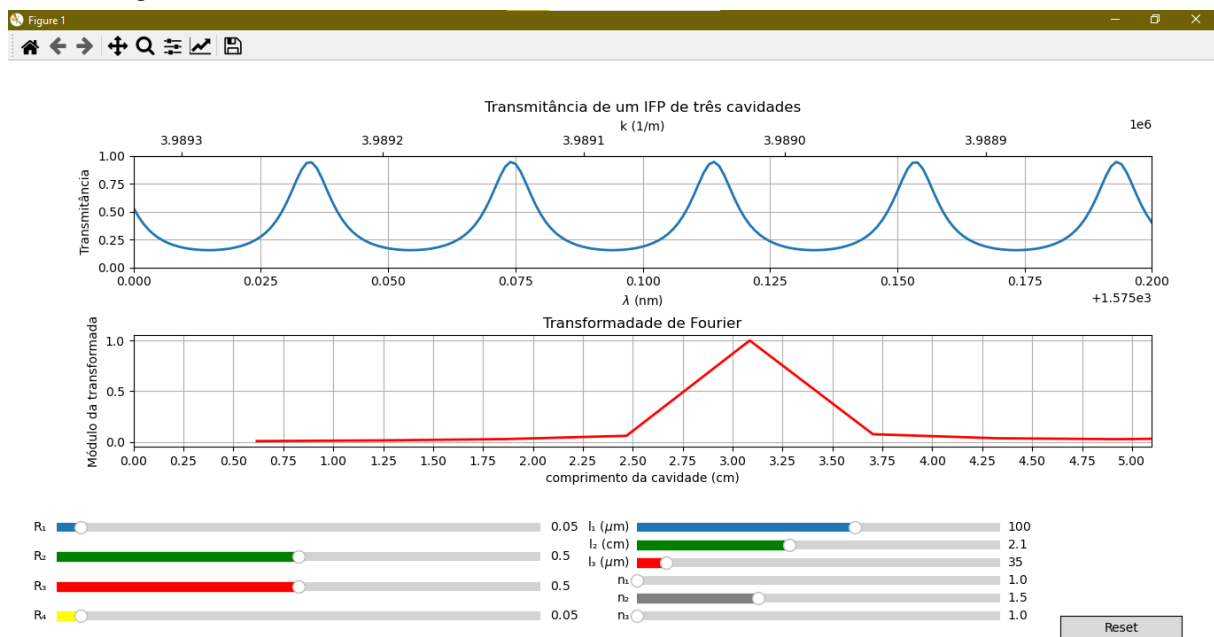
6.1 Software de Simulação

6.1.1 Funcionamento

O software para simulação de um Interferômetro de Fabry-Pérot (IFP) de 3 cavidades foi desenvolvido e seu código encontra-se no Apêndice A. A finalidade do programa foi criar a curva da transmitância T em função do comprimento de onda λ da luz que incide no IFP e acompanhar sua evolução a partir dos parâmetros do interferômetro: $R_1, R_2, R_3, R_4, l_1, l_2, l_3, n_1, n_2$ e n_3 . As refletividades R_j são configuradas a partir de controles deslizantes cujos valores variam de 0 a 1, enquanto os índices de refração, também definidos por controles deslizantes, podem assumir valores entre 1 e 3. Os parâmetros l_1 e l_3 assumem valores de 0 a $150 \mu\text{m}$ e os valores de l_2 estão compreendidos entre 0 e 5 cm . Pelo programa, torna-se possível entender a dinâmica de tais parâmetros sob o comportamento da curva T .

A Figura 13 mostra o layout do programa, que funciona do seguinte modo: a medida que o usuário altera os valores dos parâmetros nos controles deslizantes, o gráfico é atualizado e a mudança na forma da curva de T pode ser acompanhada, bem como os picos da Transformada de Fourier.

Figura 13 – Layout do programa desenvolvido para simular o comportamento da Transmitância T em função de λ para um IFP de 3 cavidades.

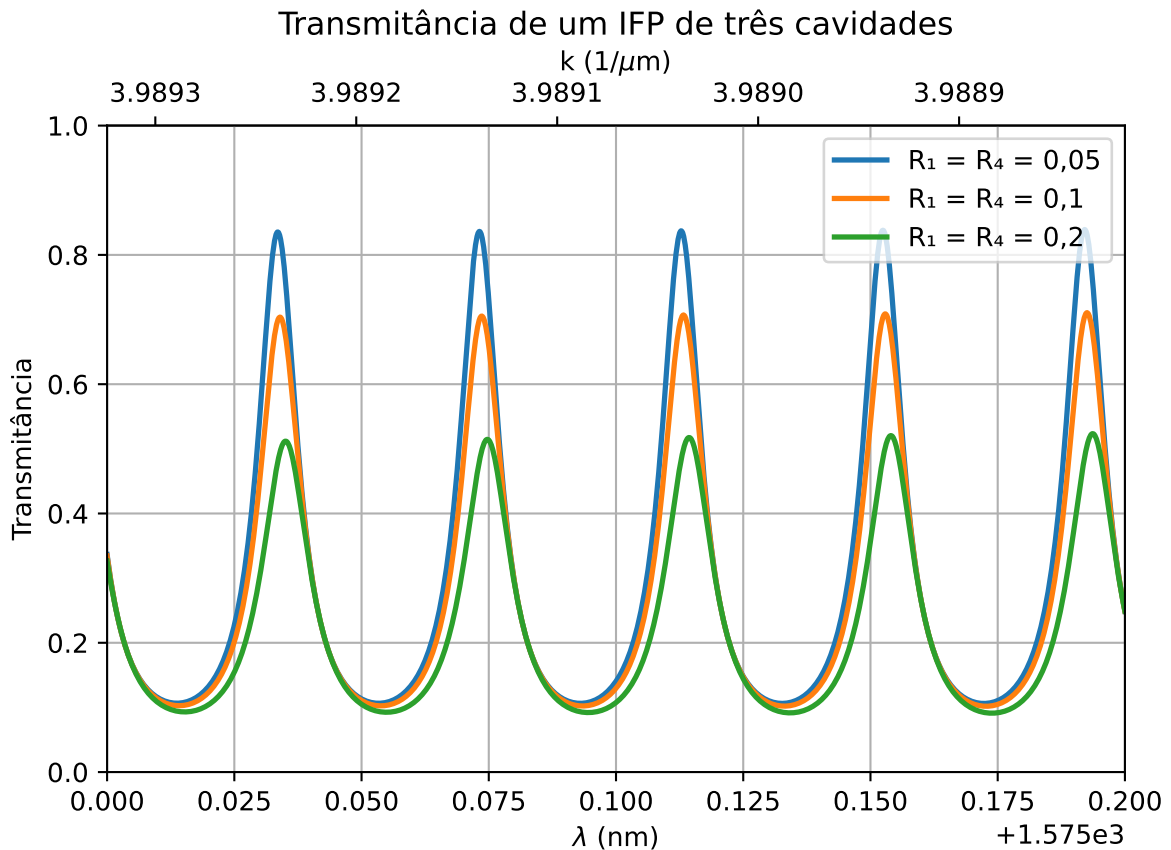


Fonte: Elaborado pelo próprio autor.

6.1.2 Simulação

Através do software desenvolvido, foi determinada a maneira como cada parâmetro influencia no espectro de transmissão. Os termos R_j determinam o grau de influência que cada cavidade causa no sinal de T . Se uma cavidade possui espelhos de refletividades R_j e R_{j+1} e estas são muito maiores que as restantes, então essa cavidade domina o sinal de T . Para os valores típicos do experimento, a cavidade da guia de onda é dominante, de forma que o espectro se aproxima muito de um IFP de uma única cavidade. As demais cavidades alteram moderadamente os valores máximos e mínimos, alterando levemente a aparência da curva que seria de uma única cavidade. A Figura 14 mostra um comparativo entre diferentes curvas quando mantém-se fixo os valores dos comprimentos e índices de refração e varia-se as refletividades R_1 e R_4 em 0,05, 0,1 e 0,2.

Figura 14 – Gráfico da transmitância T em função de λ para diferentes valores de R_1 e R_4 .

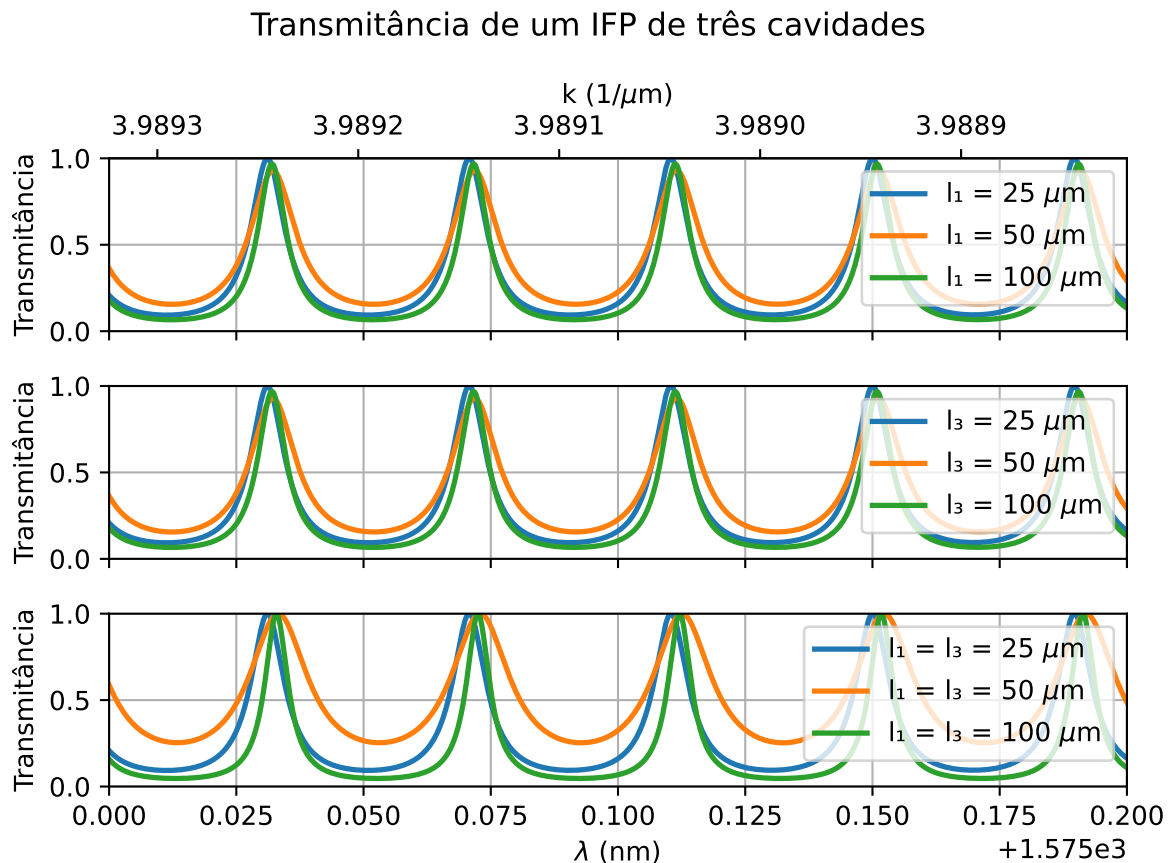


Fonte: Elaborado pelo próprio autor.

Estudando a simulação para valores típicos do experimento, os parâmetros l_1 e l_3 causam alterações consideráveis na curva T , apesar de tais termos serem muito menores que l_2 . A Figura 15 mostra como o espectro de transmissão é alterado por diferentes valores de l_1 e l_3 . Nota-se que as posições dos picos não é modificada, ou seja, a distância entre os picos (FSR - *free spectral range*) se mantém. Contudo, os máximos e mínimos do espectro são alterados, bem como o formato da curva e largura dos picos.

Em relação aos índices de refração, os termos n_1 e n_3 causam perturbação semelhante a l_1 e l_3 , uma vez que tais parâmetros compõem o comprimento óptico das cavidades e mudanças em n_j equivale a alterar os valores de l_j , isto é, mudar o comprimento óptico.

Figura 15 – Espectro de transmissão para diferentes valores de l_1 e l_3 indicados nos gráficos. No gráfico superior variou-se apenas l_1 e no gráfico do centro variou-se l_3 . Já no gráfico inferior foram variados ambos os parâmetros.



Fonte: Elaborado pelo próprio autor.

O índice de refração da guia de onda juntamente com l_2 , estabelecem o comprimento óptico da cavidade e, conseqüentemente, a frequência fundamental da função, o FSR e posições dos máximos da função. A Figura 16 apresenta as curvas do espectro de transmissão para diferentes valores de comprimento óptico da cavidade 2.

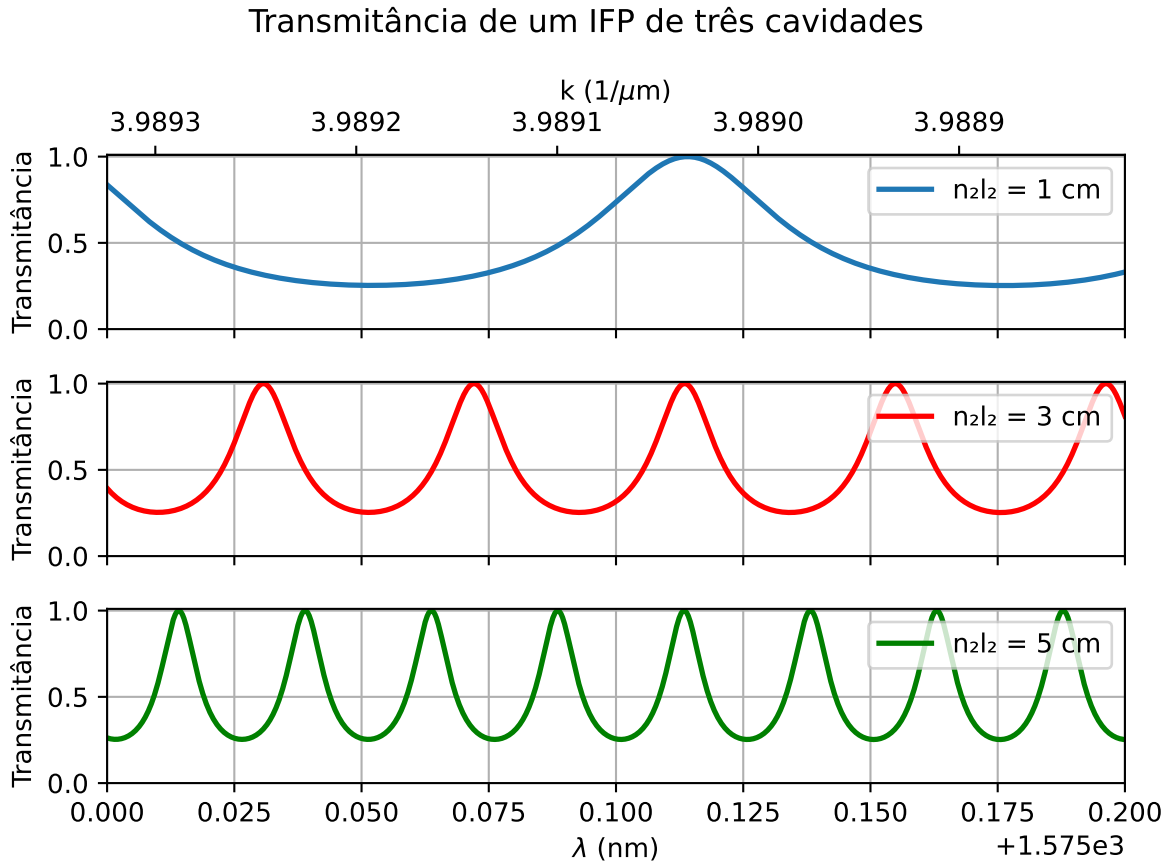
6.2 Software de Análise de dados

6.2.1 Funcionamento

O código do software desenvolvido é apresentado no Apêndice B e seu layout é mostrado na Figura 17.

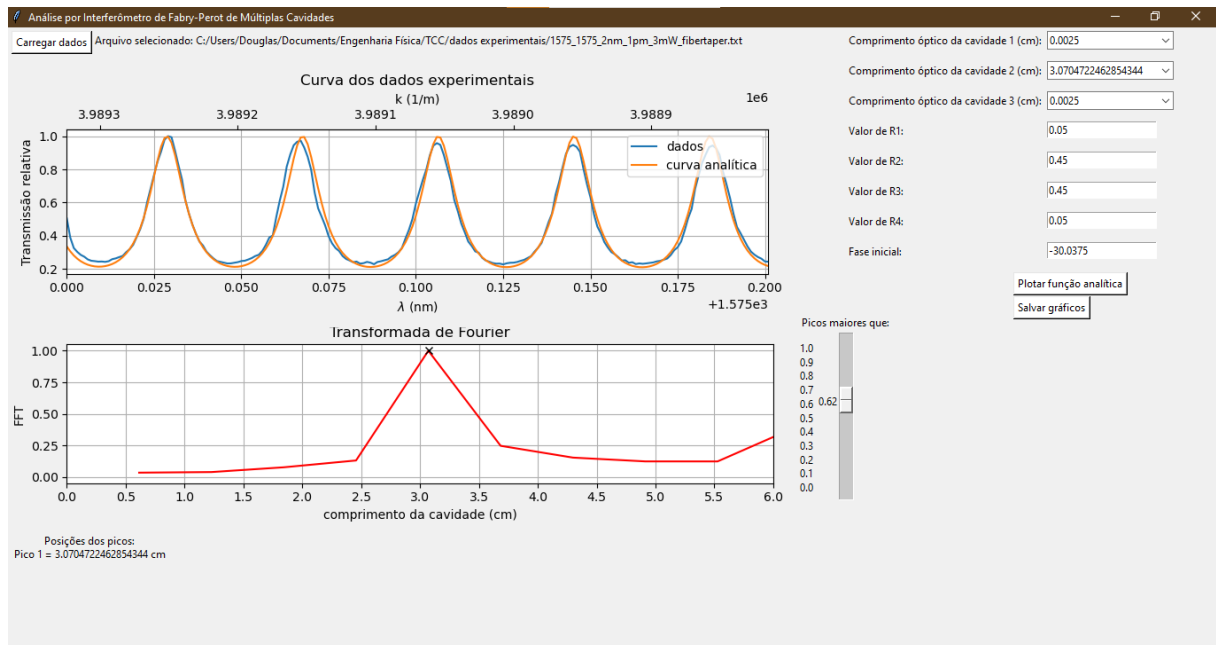
O software funciona seguindo a cadeia de comando apresentada a seguir:

Figura 16 – Gráficos do espectro de transmissão para três valores distintos do comprimento óptico da cavidade 2 (n_2l_2).



Fonte: Elaborado pelo próprio autor.

Figura 17 – Software para análise dos dados experimentais da técnica de perdas por espalhamento baseada em IFP de três cavidades.



Fonte: Elaborado pelo próprio autor.

1. Carregar os dados do experimento: o usuário clica no botão "carregar dados" e seleciona o arquivo com os dados do experimento. As informações são guardadas pelo software e a curva dos dados é plotada em um gráfico da transmitância T em função do comprimento de onda λ ou número de onda k .
2. Selecionar picos do gráfico da transformada de Fourier: O usuário seleciona a altura mínima para que o programa identifique os picos da FFT e determine suas respectivas posições.
3. Definir parâmetros: O usuário pode escolher as posições dos picos como parâmetros para os comprimentos ópticos das cavidades ou digitar outros valores quaisquer. Também são definidos os valores para as refletividades das interfaces e diferença de fase inicial do feixe.
4. Curva da função analítica: O usuário clica no botão "plotar função analítica" após todos os parâmetros serem estabelecidos para que a curva analítica seja desenhada no mesmo gráfico da curva dos dados.
5. Salvar gráficos: Os gráficos gerados no software podem ser salvos como imagens no formato JPG ao clique do botão "salvar gráficos".

6.2.2 Análise dos dados

O experimento para a medição das perdas por espalhamento fornece como dados a potência luminosa que incide no detector, ou seja, a potência que atravessou o IFP formado pela guia de onda e as cavidades entre a guia e as fibras, em função do comprimento de onda da luz. Alguns arquivos de dados foram analisados pelo software desenvolvido neste trabalho. Informações importantes das medidas dos arquivos são apresentadas na Tabela 1.

Tabela 1 – Informações das medidas presentes em cada arquivo de dados.

Arquivo	Variação do comprimento de onda λ (nm)	Resolução (pm)	Potência do laser (mW)
1	1575 - 1575.2	1	3
2	1575 - 1575.15	1	3
3	1600 - 1600.2	1	3
4	1610 - 1610.2	1	3

Fonte: Elaborado pelo próprio autor.

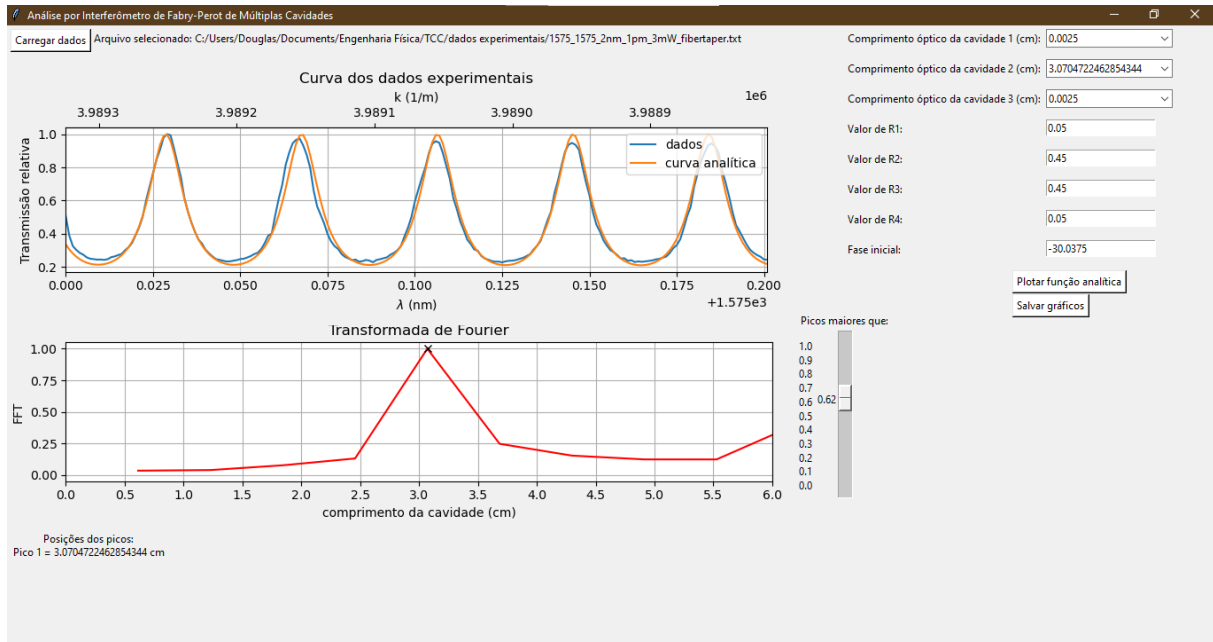
Utilizando o software, foi determinado o comprimento óptico da guia de onda ($n_j \times l_j$) dado pelo pico da transformada de Fourier. Além disso, os demais parâmetros foram inferidos, levando a curvas analíticas muito próximas às curvas dos dados experimentais. Os resultados são mostrados na Tabela 2.

Tabela 2 – Resultados dos parâmetros do IFP de três cavidades inferidos pela análise dos dados feita através do software desenvolvido no trabalho.

Arquivo	$n_1 l_1$ (cm)	$n_2 l_2$ (cm)	$n_3 l_3$ (cm)	R_1	R_2	R_3	R_4
1	0,0100	3,07	0,0035	0,05	0,45	0,45	0,05
2	0,0050	3,07	0,0050	0,05	0,35	0,35	0,05
3	0,0050	3,17	0,0050	0,05	0,49	0,49	0,05
4	0,0100	3,21	0,0100	0,05	0,47	0,47	0,05

Fonte: Elaborado pelo próprio autor.

Figura 18



Fonte: Elaborado pelo próprio autor.

7 CONCLUSÃO

REFERÊNCIAS

- ATHERTON, P. D. et al. Tunable Fabry-Perot Filters. *Optical Engineering*, SPIE, v. 20, n. 6, p. 206806, 1981. Disponível em: <<https://doi.org/10.1117/12.7972819>>. Citado na página 8.
- CHANDRASEKAR, R. et al. Photonic integrated circuits for Department of Defense-relevant chemical and biological sensing applications: state-of-the-art and future outlooks. *Optical Engineering*, SPIE, v. 58, n. 2, p. 020901, 2019. Disponível em: <<https://doi.org/10.1117/1.OE.58.2.020901>>. Citado na página 6.
- CORZINE, S. et al. Design of fabry-perot surface-emitting lasers with a periodic gain structure. *IEEE Journal of Quantum Electronics*, v. 25, n. 6, p. 1513–1524, 1989. Citado na página 8.
- COSTA, F. B. et al. Continuous wave terahertz imaging for ndt: Fundamentals and experimental validation. *Measurement*, v. 172, p. 108904, 2021. ISSN 0263-2241. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0263224120313877>>. Citado na página 11.
- CULSHAW, B. Photonics – an introduction. In: _____. *Introducing Photonics*. [S.l.]: Cambridge University Press, 2020. p. 1–10. Citado na página 6.
- DU, X. et al. A model of dual fabry-perot etalon-based external-cavity tunable laser using finite-difference traveling-wave method. *Photonics*, v. 10, n. 5, 2023. ISSN 2304-6732. Disponível em: <<https://www.mdpi.com/2304-6732/10/5/579>>. Citado na página 15.
- FABRY, C.; PÉROT, A. Sur les franges des lames minces argentées et leur application à la mesure de petites épaisseurs d'air. *Ann. Chim. Phys*, v. 12, n. 459-501, 1897. Citado 2 vezes nas páginas 8 e 15.
- FOWLES, G. *Introduction to Modern Optics*. Dover Publications, 2012. (Dover Books on Physics). ISBN 9780486134925. Disponível em: <<https://books.google.com.br/books?id=VeS7AQAQAQBAJ>>. Citado na página 6.
- HOFSTETTER, D.; THORNTON, R. L. Theory of loss measurements of fabry-perot resonators by fourier analysis of the transmission spectra. *Opt. Lett.*, Optica Publishing Group, v. 22, n. 24, p. 1831–1833, Dec 1997. Disponível em: <<https://opg.optica.org/ol/abstract.cfm?URI=ol-22-24-1831>>. Citado na página 24.
- IFTIKHAR, F.; KHAN, U.; CHEEMA, I. Digital synthesis of multistage etalons based upon unequal cavity lengths. *Journal of the Optical Society of America B*, v. 37, 04 2020. Citado na página 8.
- IRMER, S. et al. Ultralow biased widely continuously tunable fabry-perot filter. *IEEE Photonics Technology Letters*, v. 15, n. 3, p. 434–436, 2003. Citado na página 8.
- ISLAM, M. R. et al. Chronology of fabry-perot interferometer fiber-optic sensors and their applications: A review. *Sensors*, v. 14, n. 4, p. 7451–7488, 2014. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/14/4/7451>>. Citado na página 8.
- JAVALOYES, J.; BALLE, S. Mode-locking in semiconductor fabry-pérot lasers. *IEEE Journal of Quantum Electronics*, v. 46, n. 7, p. 1023–1030, 2010. Citado na página 8.

KIM, H. D.; KANG, S.-G.; LE, C.-H. A low-cost wdm source with an ase injected fabry-perot semiconductor laser. *IEEE Photonics Technology Letters*, v. 12, n. 8, p. 1067–1069, 2000. Citado na página 8.

LIU, H.; UKHTARY, M. S.; SAITO, R. Hidden symmetries in n-layer dielectric stacks. *Journal of Physics: Condensed Matter*, IOP Publishing, v. 29, n. 45, p. 455303, oct 2017. Disponível em: <<https://dx.doi.org/10.1088/1361-648X/aa865c>>. Citado 2 vezes nas páginas 9 e 11.

MÁRQUEZ-CRUZ, V. A.; HERNÁNDEZ-CORDERO, J. A. Fiber optic fabry-perot sensor for surface tension analysis. *Opt. Express*, Optica Publishing Group, v. 22, n. 3, p. 3028–3038, Feb 2014. Disponível em: <<https://opg.optica.org/oe/abstract.cfm?URI=oe-22-3-3028>>. Citado na página 9.

MÉNARD, M.; KIRK, A. G. Design of fabry-perot filters in planar waveguides with deep-etched features for spatial switching. *Opt. Express*, Optica Publishing Group, v. 17, n. 20, p. 17614–17629, Sep 2009. Disponível em: <<https://opg.optica.org/oe/abstract.cfm?URI=oe-17-20-17614>>. Citado na página 8.

MENEZES, N. *Introdução à programação com Python – 3ª edição: Algoritmos e lógica de programação para iniciantes*. Novatec Editora, 2019. ISBN 9788575227183. Disponível em: <<https://books.google.com.br/books?id=sECCDwAAQBAJ>>. Citado na página 25.

PARASHAR, A. et al. Three cavity tunable mems fabry perot interferometer. *Sensors*, v. 7, n. 12, p. 3071–3083, 2007. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/7/12/3071>>. Citado 2 vezes nas páginas 9 e 10.

PELUCCHI, E. et al. The potential and global outlook of integrated photonics for quantum technologies. *Nature Reviews Physics*, v. 4, 12 2021. Disponível em: <<https://www.nature.com/articles/s42254-021-00398-z>>. Citado na página 6.

PFEIFER, H. et al. Achievements and perspectives of optical fiber fabry–perot cavities. *Applied Physics B*, Springer Science and Business Media LLC, v. 128, n. 2, jan 2022. Disponível em: <<https://doi.org/10.1007%2Fs00340-022-07752-8>>. Citado na página 8.

SALEH, B. E. A.; TEICH, M. C. Resonator optics. In: _____. *Fundamentals of Photonics*. John Wiley Sons, Ltd, 2019. cap. 11, p. 433–472. ISBN 9781118770092. Disponível em: <<https://books.google.com.br/books?id=rcqKDwAAQBAJ>>. Citado na página 15.

SINGH, S.; DAS, S.; RAY, S. K. Progress in group-iv semiconductor nanowires based photonic devices. *Applied Physics A*, Springer, v. 129, n. 216, 2023. Disponível em: <<https://link.springer.com/article/10.1007/s00339-023-06483-7>>. Citado na página 6.

STADT, H. van de; MULLER, J. M. Multimirror fabry–perot interferometers. *J. Opt. Soc. Am. A*, Optica Publishing Group, v. 2, n. 8, p. 1363–1370, Aug 1985. Disponível em: <<https://opg.optica.org/josaa/abstract.cfm?URI=josaa-2-8-1363>>. Citado na página 8.

TAN, J. Y. et al. Applications of photonics in agriculture sector: A review. *Molecules*, v. 24, n. 10, 2019. ISSN 1420-3049. Disponível em: <<https://www.mdpi.com/1420-3049/24/10/2025>>. Citado na página 6.

TRANSPARENCY MARKET RESEARCH, INC. *Photonic Integrated Circuits (PIC) Market*. 2023. Transparency Market Research. Disponível em: <<https://www.transparencymarketresearch.com/photonic-integrated-circuit.html>>. Acesso em: 07 ago 2023. Citado na página 6.

Wikimedia Commons. *File:Etalon-1.svg*. 2009. Disponível em: <<https://commons.wikimedia.org/wiki/File:Etalon-1.svg#>>. Acesso em: 07 ago 2023. Citado na página 7.

APÊNDICE A – CÓDIGO FONTE DO PROGRAMA DE SIMULAÇÃO

O código fonte do programa desenvolvido para simular o comportamento da transmitância T do IFP de três cavidades em função do comprimento de onda λ é apresentado a seguir:

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Jun 14 10:34:22 2023
4
5 @author: Douglas
6 """
7
8 # interferometro fabry-perot 3 cavidades
9
10 ##### BIBLIOTECAS #####
11 import numpy as np
12 import matplotlib.pyplot as plt
13 from matplotlib.widgets import Slider, Button
14 from scipy.signal import find_peaks
15
16 # funcao analitica da transmissao para 3 cavidades
17 def T(x, R1, R2, R3, R4, l1, l2, l3, n1, n2, n3):
18     fi1 = 2*np.pi*n1*l1*10/x
19     fi2 = 2*np.pi*n2*l2*1e7/x
20     fi3 = 2*np.pi*n3*l3*10/x
21     D = (1 + R1*R2 + R2*R3 + R1*R3 + R3*R4 + R2*R4 + R1*R4 + R1*R2*R3*R4
22         +
23         2*(R1*R2)**0.5*(R3 + 1)*(R4 + 1)*np.cos(2*fi1) +
24         2*(R2*R3)**0.5*(R1 + 1)*(R4 + 1)*np.cos(2*fi2) +
25         2*(R1*R3)**0.5*(R4 + 1)*np.cos(2*(fi1 + fi2)) +
26         2*R2*(R1*R3)**0.5*(R4 + 1)*np.cos(2*(fi1 - fi2)) +
27         2*(R3*R4)**0.5*(R1 + 1)*(R2 + 1)*np.cos(2*fi3) +
28         4*(R1*R2*R3*R4)**0.5*np.cos(2*(fi1 + fi3)) +
29         2*(R2*R4)**0.5*(R1 + 1)*np.cos(2*(fi2 + fi3)) +
30         2*(R1*R4)**0.5*np.cos(2*(fi1 + fi2 + fi3)) +
31         4*(R1*R2*R3*R4)**0.5*np.cos(2*(fi1 - fi3)) +
32         2*R2*(R1*R4)**0.5*np.cos(2*(fi1 - fi2 - fi3)) +
33         2*R3*(R2*R4)**0.5*(R1 + 1)*np.cos(2*(fi2 - fi3)) +
34         2*R2*R3*(R1*R4)**0.5*np.cos(2*(fi1 - fi2 + fi3)) +
35         2*R3*(R1*R4)**0.5*np.cos(2*(fi1 + fi2 - fi3)))
36     return (1 - R1)*(1 - R2)*(1 - R3)*(1 - R4)/D
37
38 #variacao do comprimento de onda x

```

```
39 x_min = 1575.0 #nm
40 x_max = 1575.5 #nm
41
42
43 #PASSO
44 dx = 1e-3
45
46 # comprimento de onda
47 x = np.arange(x_min, x_max, dx)
48
49
50 # Definindo os par metros iniciais
51 l1 = l10 = 50.0 #comprimento da cavidade 1 em 10^-6 m
52 l2 = l20 = 2.1 #comprimento da cavidade 2 em micrometro
53 l3 = l30 = 50.0 #comprimento da cavidade 3 em cm
54 R2 = R3 = R20 = R30 = 0.5
55 R1 = R10 = R4 = R40 = 0.5
56 n1 = n10 = n3 = n30 = 1.0
57 n2 = n20 = 1.55 # ndice de refra o da cavidade 2
58
59 # transmiss o
60 y1 = T(x, R10, R20, R30, R40, l10, l20, l30, n10, n20, n30)
61 y1_picos, _ = find_peaks(y1)
62
63 #Aplicando a transformada de Fourier
64 yfft = np.fft.fft(y1)
65
66 # numero de onda
67 k = 2*np.pi*1e7/x
68
69 #####dk medio entre todos os pontos#####
70 diferenca_k = []
71 for i in range(0, len(k)-1):
72     diferenca = k[i] - k[i+1]
73     diferenca_k.append(diferenca)
74
75 diferenca_k.append(diferenca)
76 dk = sum(diferenca_k)/len(diferenca_k)
77 #dk2 = np.array(diferenca_k)
78
79 #frequencia fundamental em fun o de k
80 freq = np.fft.fftfreq(np.size(yfft), d=dk)
81
82 # valores positivos
83 x2 = freq[freq>0]
84 y2 = np.abs(yfft[freq>0])
85
```

```
86 #####PLOTANDO A FUNÇÃO#####
87 fig, (ax1, ax3) = plt.subplots(2, 1)
88 #ax1 = plt.subplot(221)
89 line1, = ax1.plot(x, y1, lw=2)
90 ax1.set_xlabel(r'$\lambda$ (nm)')
91 ax1.set_ylabel('Transmissão relativa')
92 ax1.set_xlim(x_min, x_max)
93 ax1.grid()
94
95
96 #####CRIAR UM SEGUNDO EIXO X NA PARTE SUPERIOR DO GRÁFICO#####
97 #####EIXO DO NÚMERO DE ONDA K#####
98 def x2k(x):
99     return 2*np.pi*1e9/x
100
101 def k2x(k):
102     return 2*np.pi/k
103
104 ax2 = ax1.secondary_xaxis('top', functions=(x2k, k2x))
105 ax2.set_xlabel('k (1/m)')
106
107 #####PLOTANDO A FFT#####
108 #ax3 = plt.subplot(222)
109 line2, = ax3.plot(x2*np.pi, y2/max(y2), 'r', lw=2)
110 ax3.grid()
111 ax3.set_xlabel(r'comprimento da cavidade (cm)')
112 ax3.set_ylabel('FFT')
113 ax3.set_xlim(0, 5.1)
114 ticks = np.arange(0, 5.1, 0.25)
115 ax3.set_xticks(ticks)
116 plt.tight_layout()
117
118 #ajustando o gráfico para a inserção dos controles deslizantes
119 fig.subplots_adjust(bottom=0.35)
120
121 #controle deslizante horizontal para R1
122 axR1 = fig.add_axes([0.04, 0.2, 0.4, 0.03])
123 R1_slider = Slider(
124     ax = axR1,
125     label = 'R\u2081',
126     valmin = 0,
127     valmax = 1,
128     valstep = 0.01,
129     valinit = R10,
130 )
131
132 #controle deslizante horizontal para R2
```

```
133 axR2 = fig.add_axes([0.04, 0.15, 0.4, 0.03])
134 R2_slider = Slider(
135     ax = axR2,
136     label = 'R\u2082',
137     valmin = 0,
138     valmax = 1,
139     valstep = 0.01,
140     valinit = R20,
141     color = 'green'
142 )
143
144 #controle deslizante horizontal para R3
145 axR3 = fig.add_axes([0.04, 0.1, 0.4, 0.03])
146 R3_slider = Slider(
147     ax = axR3,
148     label = 'R\u2083',
149     valmin = 0,
150     valmax = 1,
151     valstep = 0.01,
152     valinit = R30,
153     color = 'red'
154 )
155
156 #controle deslizante horizontal para R4
157 axR4 = fig.add_axes([0.04, 0.05, 0.4, 0.03])
158 R4_slider = Slider(
159     ax = axR4,
160     label = 'R\u2084',
161     valmin = 0,
162     valmax = 1,
163     valstep = 0.01,
164     valinit = R40,
165     color = 'yellow'
166 )
167
168 #controle deslizante horizontal para o comprimento l1 da cavidade
169 axl1 = fig.add_axes([0.52, 0.2, 0.3, 0.03])
170 l1_slider = Slider(
171     ax = axl1,
172     label = 'l\u2081 (\u03bcm)',
173     valmin = 25,
174     valmax = 150,
175     valstep = 0.1,
176     valinit = l10,
177 )
178
179
```

```
180 #controle deslizante horizontal para o comprimento l2 da cavidade
181 axl2 = fig.add_axes([0.52, 0.17, 0.3, 0.03])
182 l2_slider = Slider(
183     ax = axl2,
184     label = 'l\u2082 (cm)',
185     valmin = 0,
186     valmax = 5,
187     valstep = 0.01,
188     valinit = 120,
189     color = 'green'
190 )
191
192
193 #controle deslizante horizontal para o comprimento l3 da cavidade
194 axl3 = fig.add_axes([0.52, 0.14, 0.3, 0.03])
195 l3_slider = Slider(
196     ax = axl3,
197     label = 'l\u2083 (cm)',
198     valmin = 25,
199     valmax = 150,
200     valstep = 0.01,
201     valinit = 130,
202     color = 'red'
203 )
204
205 #controle deslizante horizontal para o n1
206 axn1 = fig.add_axes([0.52, 0.11, 0.3, 0.03])
207 n1_slider = Slider(
208     ax = axn1,
209     label = 'n\u2081',
210     valmin = 1,
211     valmax = 2.5,
212     valstep = 0.05,
213     valinit = n10,
214     color = 'gray'
215 )
216
217 #controle deslizante horizontal para o n2
218 axn2 = fig.add_axes([0.52, 0.08, 0.3, 0.03])
219 n2_slider = Slider(
220     ax = axn2,
221     label = 'n\u2082',
222     valmin = 1,
223     valmax = 2.5,
224     valstep = 0.05,
225     valinit = n20,
226     color = 'gray'
```

```
227 )
228
229 #controle deslizante horizontal para o n2
230 axn3 = fig.add_axes([0.52, 0.05, 0.3, 0.03])
231 n3_slider = Slider(
232     ax = axn3,
233     label = 'n\u2083',
234     valmin = 1,
235     valmax = 2.5,
236     valstep = 0.05,
237     valinit = n30,
238     color = 'gray'
239 )
240
241 #a fun o a ser executada sempre que o controle deslizante mudar
242 def update(val):
243     y1_2 = T(x, R1_slider.val, R2_slider.val, R3_slider.val, R4_slider.
244         val,
245             l1_slider.val, l2_slider.val, l3_slider.val, n1_slider.val,
246             n2_slider.val,
247             n3_slider.val)
248     line1.set_ydata(y1_2)
249     #line1.set_ydata(T(x, R1_slider.val, R2_slider.val, R3_slider.val,
250         R4_slider.val, l1_slider.val, l2_slider.val, l3_slider.val))
251     y2_2 = np.fft.fft(y1_2)
252     y2_2 = np.abs(y2_2)
253     y2_2 = y2_2[freq>0]
254     line2.set_ydata(y2_2/max(y2_2))
255     #line2.set_ydata(fft(x, R1_slider.val, R2_slider.val, R3_slider.val,
256         R4_slider.val, l1_slider.val, l2_slider.val, l3_slider.val)/max(fft(
257         x, R1_slider.val, R2_slider.val, R3_slider.val, R4_slider.val,
258         l1_slider.val, l2_slider.val, l3_slider.val)))
259     fig.canvas.draw_idle()
260
261 # registra a fun o de atualiza o com cada controle deslizante
262 R1_slider.on_changed(update)
263 R2_slider.on_changed(update)
264 R3_slider.on_changed(update)
265 R4_slider.on_changed(update)
266 l1_slider.on_changed(update)
267 l2_slider.on_changed(update)
268 l3_slider.on_changed(update)
269 n1_slider.on_changed(update)
270 n2_slider.on_changed(update)
271 n3_slider.on_changed(update)
272
273 # Cria um bot o de reset para redefinir os controles deslizantes.
```

```
268 resetax = fig.add_axes([0.87, 0.025, 0.1, 0.04])
269 button = Button(resetax, 'Reset', hovercolor='0.975')
270
271 def reset(event):
272     R1_slider.reset()
273     R2_slider.reset()
274     R3_slider.reset()
275     R4_slider.reset()
276     l1_slider.reset()
277     l2_slider.reset()
278     l3_slider.reset()
279     n1_slider.reset()
280     n2_slider.reset()
281     n3_slider.reset()
282 button.on_clicked(reset)
283
284 plt.show()
```

APÊNDICE B – CÓDIGO FONTE DO SOFTWARE DE ANÁLISE DOS DADOS EXPERIMENTAIS

O código fonte do software de análise dos dados experimentais da técnica de medição de perdas por espalhamento é apresentado a seguir:

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Jul 25 09:50:30 2023
4
5 @author: Douglas
6 """
7
8 import numpy as np
9 from tkinter import *
10 from tkinter import filedialog as fd
11 from tkinter import ttk
12 from tkinter import messagebox
13 import matplotlib.pyplot as plt
14 from matplotlib.figure import Figure
15 from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
16 from scipy.signal import find_peaks
17 import matplotlib.backends.backend_tkagg as tkagg
18
19 #import pandas as pd
20
21 erro = 1
22
23 # variáveis iniciais
24 x_dados = y_dados = 0
25 plot1 = plot2 = 0
26 picos = picos_freq = 0
27 R1_input = R2_input = R3_input = R4_input = n_input = 0
28
29 # Definindo os parâmetros iniciais
30 l1 = l10 = 50.0          #comprimento da cavidade 1 em micrometro
31 l2 = l20 = 3.20841      #comprimento da cavidade 2 em cm
32 l3 = l30 = 50.0        #comprimento da cavidade 3 em micrometro
33 R2 = R3 = R20 = R30 = 0.45
34 R1 = R10 = R4 = R40 = 0.05
35 n = n0 = 1.0          # ndice de refração da cavidade 2
36
37
38 # função analítica da transmissão para 3 cavidades
39 def T(x, R1, R2, R3, R4, l1, l2, l3, n):

```

```

40     fi1 = 2*np.pi*l1*1e7/x
41     fi2 = 2*np.pi*n*l2*1e7/x
42     fi3 = 2*np.pi*l3*1e7/x
43     D = (1 + R1*R2 + R2*R3 + R1*R3 + R3*R4 + R2*R4 + R1*R4 + R1*R2*R3*R4
44         +
45         2*(R1*R2)**0.5*(R3 + 1)*(R4 + 1)*np.cos(2*fi1) +
46         2*(R2*R3)**0.5*(R1 + 1)*(R4 + 1)*np.cos(2*fi2) +
47         2*(R1*R3)**0.5*(R4 + 1)*np.cos(2*(fi1 + fi2)) +
48         2*R2*(R1*R3)**0.5*(R4 + 1)*np.cos(2*(fi1 - fi2)) +
49         2*(R3*R4)**0.5*(R1 + 1)*(R2 + 1)*np.cos(2*fi3) +
50         4*(R1*R2*R3*R4)**0.5*np.cos(2*(fi1 + fi3)) +
51         2*(R2*R4)**0.5*(R1 + 1)*np.cos(2*(fi2 + fi3)) +
52         2*(R1*R4)**0.5*np.cos(2*(fi1 + fi2 + fi3)) +
53         4*(R1*R2*R3*R4)**0.5*np.cos(2*(fi1 - fi3)) +
54         2*R2*(R1*R4)**0.5*np.cos(2*(fi1 - fi2 - fi3)) +
55         2*R3*(R2*R4)**0.5*(R1 + 1)*np.cos(2*(fi2 - fi3)) +
56         2*R2*R3*(R1*R4)**0.5*np.cos(2*(fi1 - fi2 + fi3)) +
57         2*R3*(R1*R4)**0.5*np.cos(2*(fi1 + fi2 - fi3)))
58     return (1 - R1)*(1 - R2)*(1 - R3)*(1 - R4)/D
59
60 def inicio():
61     global fig1, fig2
62     global plot1, plot2
63     global canvas1, canvas2
64
65     # gr fico dos dados
66     fig1 = Figure(figsize=(8.9, 3), dpi=100, tight_layout=True,
67 facecolor='#f0f0f0')
68     plot1 = fig1.add_subplot(111)
69     plot1.set_xlabel(r'$\lambda$ (nm)')
70     plot1.set_ylabel('Transmiss o relativa')
71     plot1.set_title('Curva dos dados experimentais')
72     plot1.grid()
73     plot1.set_xlim(1500, 1600)
74
75     # EIXO X SECUND RIO NA PARTE SUPERIOR DO GR FICO
76     # fun es de convers o entre n mero de onda e comprimento de
77     onda
78     def x2k(x):
79         return 2*np.pi*1e9/x
80
81     def k2x(k):
82         return 2*np.pi/k
83
84     eixo2 = plot1.secondary_xaxis('top', functions=(x2k, k2x))
85     eixo2.set_xlabel('k (1/m)')

```

```
84
85     canvas1 = FigureCanvasTkAgg(fig1, janela)
86     canvas1.get_tk_widget().place(x=0, y='1c')
87
88     # gráfico da transformada de fourier
89     fig2 = Figure(figsize=(8.9, 2.5), dpi=100, tight_layout=True,
90     facecolor='#f0f0f0')
91     plot2 = fig2.add_subplot(111)
92     plot2.set_xlabel(r'comprimento da cavidade (cm)')
93     plot2.set_ylabel('FFT')
94     plot2.set_title('Transformada de Fourier')
95     plot2.grid()
96     plot2.set_xlim(0, 6)
97
98     canvas2 = FigureCanvasTkAgg(fig2, janela)
99     canvas2.get_tk_widget().place(x=0, y='8.5c')
100
101 # função para carregar os dados
102 def carregar_arquivo():
103     global x_dados, y_dados
104     global fig1, fig2
105     global plot1, plot2
106     global picos, picos_y, picos_freq, y_norm, freq_positivas
107     global canvas1, canvas2
108     global erro
109
110     arquivo = fd.askopenfilename(initialdir = 'C:/Users/Douglas/
111     Documents/Engenharia Física/TCC/dados experimentais', title = '
112     Selecione o arquivo',
113     filetypes = [('Arquivos de Texto', '*.
114     txt'), ('Todos os arquivos', '*.*')])
115
116     # Verifica se um arquivo foi selecionado
117     if arquivo:
118         # Inicializa duas listas vazias para armazenar as colunas de
119         # n meros
120         coluna1 = []
121         coluna2 = []
122
123         # Lê todas as linhas do arquivo e separa os n meros nas listas
124         # de colunas
125         with open(arquivo, "r") as arquivo_aberto:
126             for linha in arquivo_aberto:
127                 dados = linha.strip().split()
128                 coluna1.append(float(dados[0]))
129                 coluna2.append(float(dados[1]))
```

```
125
126     # Exibe o caminho do arquivo selecionado na janela
127     label_arquivo.config(text=f"Arquivo selecionado: {arquivo}")
128
129     # Cria arrays a partir das listas de colunas
130     x_dados = np.array(coluna1)      #comprimento de onda
131     y_dados = np.array(coluna2)      #intensidade em mW
132
133     # Exibe os arrays de colunas no terminal
134     print("Comprimento de onda:")
135     print(x_dados)
136     print("Intensidade:")
137     print(y_dados)
138
139     # Transformada de Fourier dos dados
140     y_fourier = np.fft.fft(y_dados)
141
142     # N mero de onda
143     k = 2*np.pi*1e7/x_dados #[1/cm]
144
145     # dk m dio entre todos os pontos
146     diferenca_k = []
147     for i in range(0, len(k)-1):
148         diferenca = k[i] - k[i+1]
149         diferenca_k.append(diferenca)
150
151     dk = sum(diferenca_k)/len(diferenca_k)
152
153     #frequencia fundamental da fun o em fun o de k
154     freq = np.fft.fftfreq(np.size(y_fourier), d=dk)
155
156     # frequencias positivas
157     freq_positivas = freq[freq>(0/np.pi)]
158
159     # valores absolutos da transformada para frequencias positivas
160     y_fourier_abs = np.abs(y_fourier[freq>(0/np.pi)])
161     y_norm = y_fourier_abs/max(y_fourier_abs)
162
163     # gr fico dos dados
164     fig1 = Figure(figsize=(8.9, 3), dpi=100, tight_layout=True,
165 facecolor='#f0f0f0')
166     plot1 = fig1.add_subplot(111)
167     plot1.plot(x_dados, y_dados/max(y_dados))
168     plot1.set_xlabel(r'$\lambda$ (nm)')
169     plot1.set_ylabel('Transmiss o relativa')
170     plot1.set_title('Curva dos dados experimentais')
171     plot1.grid()
```

```
171     plot1.set_xlim(x_dados[0], x_dados[-1])
172
173
174     # EIXO X SECUNDARIO NA PARTE SUPERIOR DO GRFICO
175     # fun es de convers o entre n mero de onda e comprimento de
onda
176     def x2k(x):
177         return 2*np.pi*1e9/x
178
179     def k2x(k):
180         return 2*np.pi/k
181
182     eixo2 = plot1.secondary_xaxis('top', functions=(x2k, k2x))
183     eixo2.set_xlabel('k (1/m)')
184
185     canvas1 = FigureCanvasTkAgg(fig1, janela)
186     canvas1.get_tk_widget().place(x=0, y='1c')
187
188     # gr fico da transformada de fourier
189     fig2 = Figure(figsize=(8.9, 2.5), dpi=100, tight_layout=True,
facecolor='#f0f0f0')
190     plot2 = fig2.add_subplot(111)
191     plot2.plot(freq_positivas*np.pi, y_norm, 'r')
192     plot2.set_xlabel(r'comprimento da cavidade (cm)')
193     plot2.set_ylabel('FFT')
194     plot2.set_title('Transformada de Fourier')
195     plot2.grid()
196     plot2.set_xlim(0, 6)
197     ticks = np.arange(0, 6.1, 0.5)
198     plot2.set_xticks(ticks)
199
200     canvas2 = FigureCanvasTkAgg(fig2, janela)
201     canvas2.get_tk_widget().place(x=0, y='8.5c')
202
203     # texto que mostra os picos
204     texto = 'Posi es dos picos:'
205     label_picos.config(text=texto)
206
207     erro = 0
208
209     else:
210         label_arquivo.config(text="Nenhum arquivo selecionado.")
211         erro = 1
212
213
214 def analitica():
215     global fig1
```

```

216     global plot1, plot2
217     global canvas1, canvas2
218
219     if erro==0:
220         l1 = float(l1_input.get())
221         l2 = float(l2_input.get())
222         l3 = float(l3_input.get())
223         R1 = float(R1_input.get())
224         R2 = float(R2_input.get())
225         R3 = float(R3_input.get())
226         R4 = float(R4_input.get())
227         n = float(n_input.get())
228
229         fig1 = Figure(figsize=(8.9, 3), dpi=100, tight_layout=True,
facecolor='#f0f0f0')
230         plot1 = fig1.add_subplot(111)
231         plot1.plot(x_dados, y_dados/max(y_dados))
232         plot1.set_xlabel(r'$\lambda$ (nm)')
233         plot1.set_ylabel('Transmissão relativa')
234         plot1.set_title('Curva dos dados experimentais')
235         plot1.grid()
236         plot1.set_xlim(x_dados[0], x_dados[-1])
237
238         # EIXO X SECUNDÁRIO NA PARTE SUPERIOR DO GRÁFICO
239         # função de conversão entre número de onda e comprimento de
onda
240         def x2k(x):
241             return 2*np.pi*1e9/x
242
243         def k2x(k):
244             return 2*np.pi/k
245
246         eixo2 = plot1.secondary_xaxis('top', functions=(x2k, k2x))
247         eixo2.set_xlabel('k (1/m)')
248
249         canvas1 = FigureCanvasTkAgg(fig1, janela)
250         canvas1.get_tk_widget().place(x=0, y='1c')
251
252         plot1.plot(x_dados, T(x_dados, R1, R2, R3, R4, l1, l2, l3, n))
253
254     else:
255         messagebox.showerror('Erro', 'Não existem dados.')
256
257 def plotar_picos(event):
258     global fig2, plot2
259     global picos, picos_y, picos_freq, y_norm, freq_positivas, alt_pico
260

```

```
261 alt = alt_pico.get()
262 picos = find_peaks(y_norm, height=(alt))
263 picos_y = y_norm[picos[0]]
264 picos_freq = freq_positivas[picos[0]]*np.pi
265
266 if y_norm[0]==max(y_norm):
267     pico1 = y_norm[0]
268     picos_y = np.append(pico1, picos_y)
269     freq1 = freq_positivas[0]*np.pi
270     picos_freq = np.append(freq1, picos_freq)
271
272
273 fig2 = Figure(figsize=(8.9, 2.5), dpi=100, tight_layout=True,
facecolor='#f0f0f0')
274 plot2 = fig2.add_subplot(111)
275 plot2.plot(freq_positivas*np.pi, y_norm, 'r')
276 plot2.set_xlabel(r'comprimento da cavidade (cm)')
277 plot2.set_ylabel('FFT')
278 plot2.set_title('Transformada de Fourier')
279 plot2.grid()
280 plot2.set_xlim(0, 6)
281 ticks = np.arange(0, 6.1, 0.5)
282 plot2.set_xticks(ticks)
283
284 canvas2 = FigureCanvasTkAgg(fig2, janela)
285 canvas2.get_tk_widget().place(x=0, y='8.5c')
286
287 plot2.plot(picos_freq, picos_y, 'x', color='k')
288
289 texto = 'Posi es dos picos:'
290 for i in range(0, len(picos_freq)):
291     texto = texto + '\n' + f'Pico {i+1} = ' + str(picos_freq[i]) + '
cm'
292
293 label_picos.config(text=texto)
294
295 # atualizar as op es para os valores dos comprimentos
296 caixa_l1['values'] = list(picos_freq)
297 caixa_l2['values'] = list(picos_freq)
298 caixa_l3['values'] = list(picos_freq)
299
300
301 def validate_float_input(action, value_if_allowed):
302     if action == '1': # Verifica se uma inser o (digitar)
303         try:
304             float(value_if_allowed)
305             return True
```

```
306         except ValueError:
307             return False
308     return True
309
310 def on_validate_float_input(P):
311     if P:
312         try:
313             float(P)
314             return True
315         except ValueError:
316             messagebox.showerror("Erro", "Digite um número válido.")
317             return False
318     else:
319         return True
320
321 def on_combobox_select(event):
322     selected_value = float(combobox_var.get()) # Converter para float
323     print("Valor selecionado:", selected_value)
324
325
326 def salvar_graficos():
327     if erro==0:
328         imagem = fd.asksaveasfilename(initialdir = '/',title = 'Salvar
329         como',filetypes = (("jpg files","*.jpg"),("todos os arquivos","*.*"))
330         )
331         if imagem!="":
332             imagem1 = imagem + '_dados.jpg'
333             imagem2 = imagem + '_fourier.jpg'
334             fig1.savefig(imagem1)
335             fig2.savefig(imagem2)
336             messagebox.showinfo('Informação', 'Os gráficos foram
337             salvos corretamente.')
338         else:
339             messagebox.showwarning('Alerta', 'Os gráficos não foram
340             salvos.')
341     else:
342         messagebox.showerror('Erro', 'Não existem dados.')
343
344 # Interface gráfica
345 janela = Tk() #cria a janela
346 janela.title('Análise por Interferômetro de Fabry-Perot de Múltiplas
347         Cavidades')
```

```
348 botoa1 = Button(janela, text='Carregar dados', bg='white', command=
    carregar_arquivo)
349 botoa1.place(x='1m', y='1m')
350
351 # Label para exibir o caminho do arquivo selecionado
352 label_arquivo = Label(janela, text="Nenhum arquivo selecionado.")
353 #label_arquivo.grid(column=1, row=0)
354 label_arquivo.place(x='2.5c', y='1m')
355
356 # Label para exibir os comprimentos da cavidade
357 label_picos = Label(janela, text='''Posi es dos picos:''')
358 label_picos.place(x='1m', y='15c')
359
360 # Caixa para escolha do L1
361 label_l1 = Label(janela, text='Comprimento da cavidade 1 (cm):')
362 label_l1.place(x='25c', y='1m')
363 l1_input = StringVar()
364 caixa_l1 = ttk.Combobox(janela, values = picos_freq, textvariable=
    l1_input)
365 caixa_l1.place(x='30c', y='1m')
366
367 # Caixa para escolha do L2
368 label_l2 = Label(janela, text='Comprimento da cavidade 2 (cm):')
369 label_l2.place(x='25c', y='1c')
370 l2_input = StringVar()
371 caixa_l2 = ttk.Combobox(janela, values = picos_freq, textvariable=
    l2_input)
372 caixa_l2.place(x='30c', y='1c')
373
374 # Caixa para escolha do L3
375 label_l3 = Label(janela, text='Comprimento da cavidade 3 (cm):')
376 label_l3.place(x='25c', y='1.9c')
377 l3_input = StringVar()
378 caixa_l3 = ttk.Combobox(janela, values = picos_freq, textvariable=
    l3_input)
379 caixa_l3.place(x='30c', y='1.9c')
380
381 # input para o valor de R1
382 R1_input = StringVar()
383 label_R1 = Label(janela, text='Valor de R1: ')
384 label_R1.place(x='25c', y='2.8c')
385 validate_float = janela.register(validate_float_input)
386 entrada1 = Entry(janela, validate='key', validatecommand=(validate_float
    , '%d', '%P'), textvariable=R1_input)
387 entrada1.config(validatecommand=(entrada1.register(
    on_validate_float_input), '%P'))
388 entrada1.place(x='30c', y='2.8c')
```

```
389
390 # input para o valor de R2
391 R2_input = StringVar()
392 label_R2 = Label(janela, text='Valor de R2: ')
393 label_R2.place(x='25c', y='3.7c')
394 validate_float = janela.register(validate_float_input)
395 entrada2 = Entry(janela, validate='key', validatecommand=(validate_float
    , '%d', '%P'), textvariable=R2_input)
396 entrada2.config(validatecommand=(entrada1.register(
    on_validate_float_input), '%P'))
397 entrada2.place(x='30c', y='3.7c')
398
399 # input para o valor de R3
400 R3_input = StringVar()
401 label_R3 = Label(janela, text='Valor de R3: ')
402 label_R3.place(x='25c', y='4.6c')
403 validate_float = janela.register(validate_float_input)
404 entrada3 = Entry(janela, validate='key', validatecommand=(validate_float
    , '%d', '%P'), textvariable=R3_input)
405 entrada3.config(validatecommand=(entrada1.register(
    on_validate_float_input), '%P'))
406 entrada3.place(x='30c', y='4.6c')
407
408 # input para o valor de R4
409 R4_input = StringVar()
410 label_R4 = Label(janela, text='Valor de R4: ')
411 label_R4.place(x='25c', y='5.5c')
412 validate_float = janela.register(validate_float_input)
413 entrada4 = Entry(janela, validate='key', validatecommand=(validate_float
    , '%d', '%P'), textvariable=R4_input)
414 entrada4.config(validatecommand=(entrada1.register(
    on_validate_float_input), '%P'))
415 entrada4.place(x='30c', y='5.5c')
416
417 # input para o valor de n
418 n_input = StringVar()
419 label_n = Label(janela, text=' ndice de refra o da cavidade 2: ')
420 label_n.place(x='25c', y='6.4c')
421 validate_float = janela.register(validate_float_input)
422 entrada5 = Entry(janela, validate='key', validatecommand=(validate_float
    , '%d', '%P'), textvariable=n_input)
423 entrada5.config(validatecommand=(entrada1.register(
    on_validate_float_input), '%P'))
424 entrada5.place(x='30c', y='6.4c')
425
426 # Bot o para plotar a fun o anal tica
```

```
427 botao2 = Button(janela, text='Plotar fun  o anal tica', bg='white',
    command=analitica)
428 botao2.place(x='30c', y='7.3c')
429
430 #controle deslizante para a altura do pico
431 label_altura = Label(janela, text='Picos maiores que:')
432 label_altura.place(x='23.6c', y='8.5c')
433 alt_pico = DoubleVar()
434 altura = Scale(janela, orient=VERTICAL, variable=alt_pico, length='5c',
    from_=1.0, to=0.0,
435                 resolution=0.01, tickinterval=0.1)
436 altura.place(x='23.5c', y='9c')
437 altura.set(1.0)
438 # Associar a fun  o de mudan a ao evento '<<SliderChanged>>'
439 altura.bind("<ButtonRelease-1>", plotar_picos)
440
441 # Bot  o para salvar os gr ficos
442 botao3 = Button(janela, text='Salvar gr ficos', bg='white', command=
    salvar_graficos)
443 botao3.place(x='30c', y='8c')
444
445 # Execu  o da interface gr fica
446 janela.mainloop() #deixa a janela aberta
```

ANEXO A – TÍTULO DO ANEXO A

Sed mattis, erat sit amet gravida malesuada, elit augue egestas diam, tempus scelerisque nunc nisl vitae libero. Sed consequat feugiat massa. Nunc porta, eros in eleifend varius, erat leo rutrum dui, non convallis lectus orci ut nibh. Sed lorem massa, nonummy quis, egestas id, condimentum at, nisl. Maecenas at nibh. Aliquam et augue at nunc pellentesque ullamcorper. Duis nisl nibh, laoreet suscipit, convallis ut, rutrum id, enim. Phasellus odio. Nulla nulla elit, molestie non, scelerisque at, vestibulum eu, nulla. Ut odio nisl, facilisis id, mollis et, scelerisque nec, enim. Aenean sem leo, pellentesque sit amet, scelerisque sit amet, vehicula pellentesque, sapien.

ANEXO B – TÍTULO DO ANEXO B

Sed consequat tellus et tortor. Ut tempor laoreet quam. Nullam id wisi a libero tristique semper. Nullam nisl massa, rutrum ut, egestas semper, mollis id, leo. Nulla ac massa eu risus blandit mattis. Mauris ut nunc. In hac habitasse platea dictumst. Aliquam eget tortor. Quisque dapibus pede in erat. Nunc enim. In dui nulla, commodo at, consectetur nec, malesuada nec, elit. Aliquam ornare tellus eu urna. Sed nec metus. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

ANEXO C – TÍTULO DO ANEXO C

Phasellus id magna. Duis malesuada interdum arcu. Integer metus. Morbi pulvinar pellentesque mi. Suspendisse sed est eu magna molestie egestas. Quisque mi lorem, pulvinar eget, egestas quis, luctus at, ante. Proin auctor vehicula purus. Fusce ac nisl aliquam ante hendrerit pellentesque. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Morbi wisi. Etiam arcu mauris, facilisis sed, eleifend non, nonummy ut, pede. Cras ut lacus tempor metus mollis placerat. Vivamus eu tortor vel metus interdum malesuada.