



UNIVERSITA' POLITECNICA DELLE MARCHE
FACOLTA' DI INGEGNERIA

Corso di Laurea triennale Ingegneria Biomedica

**UN CODICE PHYTON PER L'OTTIMIZZAZIONE DI UNO
SCHERMO ELETTROMAGNETICO STRATIFICATO PER
RIDURRE AL MINIMO L'ESPOSIZIONE UMANA**

**A PHYTON CODE FOR THE OPTIMIZATION OF A LAYERED
ELECTROMAGNETIC SHIELD TO MINIMIZE THE HUMAN
EXPOSURE**

Relatore:

Prof. Franco Moglie

Tesi di Laurea di:

Lisa Trovarelli

Correlatore:

Prof. Valter Mariani Primiani

A.A 2021/2022

*Alla mia famiglia, che da
sempre mi sostiene e crede in me.*

Al nonno Giustino e alla nonna Paola.

INDICE

INTRODUZIONE	5
CAPITOLO 1 – Richiami di elettromagnetismo	7
1.1 Equazioni di Maxwell	7
1.2 Onde Piane	9
1.3 Propagazione onde piane in mezzi con perdite	11
1.4 Schermatura elettromagnetica	13
1.5 Efficacia di schermatura	15
1.6 Analisi in frequenza della SE	18
CAPITOLO 2 – Efficacia di schermatura per un vetro AGC	20
2.1 Analisi del vetro	20
2.2 Descrizione del modello	22
2.3 Introduzione a Python	23
2.4 Algoritmo per il calcolo della SE	25
2.4.1 Importazione delle librerie	25
2.4.2 Funzione per il calcolo dei parametri	26
2.4.3 Leggi di Snell	27
2.4.4 Polarizzazione perpendicolare	28
2.4.5 Funzione per il calcolo del campo elettrico in ogni strato	30
2.4.6 Funzione per il calcolo della SE	31
2.4.7 Main (prima dell'ottimizzazione)	33
2.4.8 SE medio	35

CAPITOLO 3 – Ottimizzazione	37
3.1 Funzione SE.....	37
3.2 Calcolo dell’SE dopo l’ottimizzazione	38
3.3 SE medio dopo l’ottimizzazione	41
Conclusioni	43
Bibliografia	44
Ringraziamenti	45

Introduzione

In tutti gli ambienti, domestici o di lavoro sono presenti campi elettromagnetici che possono avere origine naturale come la luce stessa, o artificiale quindi generati dalla presenza di impianti e dispositivi elettrici. Nel corso del ventesimo secolo, l'esposizione ambientale a campi elettromagnetici è costantemente aumentata a seguito della richiesta d'energia e del continuo sviluppo delle tecnologie di comunicazione senza fili. Quando un essere umano è immerso in un campo elettromagnetico, esso assorbe energia e si ha uno scostamento dalle condizioni di equilibrio naturale, per questo è importante proteggere le persone dai possibili effetti a lungo termine provocati dall'assorbimento di onde elettromagnetiche sul corpo umano. L'interesse al problema dell'inquinamento elettromagnetico è cresciuto soprattutto negli ultimi anni, a seguito delle numerose ricerche e studi eseguiti. Sono state redatte norme e documenti tecnici, fino alla promulgazione di leggi specifiche a tutela degli ambienti maggiormente a rischio: questo rischio deve essere valutato e limitato per garantire la non pericolosità dell'ambiente e dell'attività svolta. Oltre ai possibili effetti sul corpo umano i campi elettromagnetici possono creare disturbi ed interferenze con le apparecchiature elettroniche, per questo sono stati fissati dei limiti, a livello europeo, che ne garantiscono un rendimento soddisfacente e sicuro. In tutti i processi industriali che richiedono correnti di elevata intensità o l'impiego di intensi campi elettrici o magnetici, l'utilizzo di appositi sistemi di schermatura è fondamentale per proteggere sia i lavoratori sia le apparecchiature elettroniche presenti nelle vicinanze delle sorgenti di campo.

Tutti gli effetti conosciuti dovuti ad esposizioni del corpo umano a campi elettrici e magnetici variabili nel tempo sono da associare all'induzione di campi e correnti all'interno dell'organismo esposto. I campi elettrici esercitano delle forze su qualsiasi particella caricata elettricamente come possono essere gli ioni nei liquidi. Di conseguenza tutte le particelle che vengono investite da un campo elettrico si muovono fino a raggiungere una disposizione superficiale di equilibrio elettrostatico tale per cui all'interno del corpo umano il campo è nullo. Nel caso in cui il campo elettrico sia variabile nel tempo le cariche modificano la loro posizione in funzione del segno del campo cercando continuamente di raggiungere l'equilibrio, creando di conseguenza un moto di cariche sulla superficie (corrente elettrica indotta dal campo elettrico variabile) che aumenta di intensità all'aumentare della frequenza con cui varia il campo inducente. In presenza di campo magnetico variabile nel tempo, invece, si attiva un meccanismo diverso: questo campo genera nello spazio circostante un campo elettrico variabile nel tempo.

Se il campo elettrico variabile viene prodotto direttamente all'interno del corpo umano, genera una corrente elettrica: mentre il campo elettrico come fonte principale genera correnti superficiali al corpo, il campo magnetico provoca la circolazione di correnti all'interno del corpo stesso interessando parti molto più delicate. L'induzione di campi e correnti elettriche all'interno del corpo umano dà quindi luogo ad effetti biologici che possono essere potenziali cause di effetti sanitari ed effetti termici connessi al riscaldamento per effetto Joule.

Pertanto, sono state elaborate adeguate protezioni, definite schermi: sono costituiti da una barriera metallica che racchiude all'interno un apparecchio elettronico, che limita la propagazione delle onde elettromagnetiche per proteggere il dispositivo stesso dalle emissioni esterne e, contemporaneamente, proteggere gli apparati o esseri umani situati nelle sue vicinanze.

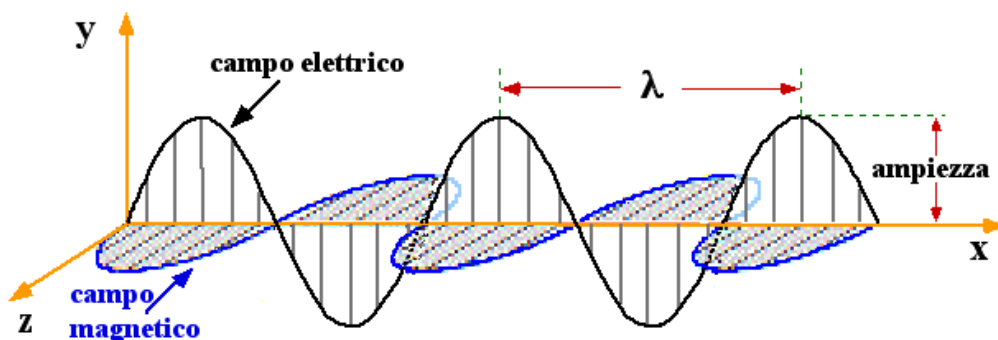
In questa tesi verrà illustrata la valutazione dell'efficacia di schermatura di un sistema formato da un'unità a doppio vetro, prodotta dall'AGC, all'interno della quale viene inserita una sottile lamina di metallo che avrà proprio la funzione di barriera schermante. In particolare, è stato creato un codice in linguaggio Python (con l'utilizzo di alcune sue librerie esterne) per valutare l'SE di tale sistema quando su di esso incide un'onda elettromagnetica (in modo obliquo) e polarizzata perpendicolarmente.

L'elaborato è organizzato nel seguente modo: inizialmente è presente una panoramica generale sui concetti fondamentali su cui si basa l'intero studio, quindi richiami sui campi elettromagnetici, la propagazione delle onde piane, l'introduzione agli schermi elettromagnetici e in particolare all'efficacia di schermatura. In seguito, vengono descritti i vetri presi in esame, una breve descrizione del modello e un'introduzione al linguaggio Python utilizzato per la creazione dell'intero codice. Infine, vi è la presentazione del codice stesso, con una descrizione generale delle varie funzioni che sono state implementate per la valutazione dell'SE, e l'esposizione dei risultati ottenuti tramite la simulazione.

CAPITOLO 1 – Richiami di elettromagnetismo

1.1 Equazioni di Maxwell

Con elettromagnetismo si identifica quella branca della fisica che si occupa dei fenomeni di natura elettrica e magnetica e delle loro correlazioni. L'elettromagnetismo studia infatti l'interazione tra campo elettrico e campo magnetico attraverso le celebri equazioni di Maxwell. Nella seconda metà dell'800 infatti, il fisico James Clerk Maxwell riuscì a riassumere i fenomeni elettrici e magnetici osservati fino ad allora in quattro equazioni differenziali. La teoria dell'elettromagnetismo che ne conseguì diede luogo ad uno dei più importanti risultati scientifici di tutti i tempi: la scoperta delle onde elettromagnetiche.



Le cariche elettriche inducono campi elettrici, mentre le correnti elettriche inducono campi magnetici. Tuttavia, la conseguenza di cariche e correnti che variano nel tempo non è soltanto la variazione corrispondente dei campi: i campi elettrici e magnetici diventano interconnessi e il loro accoppiamento genera onde elettromagnetiche in grado di propagarsi attraverso lo spazio libero e nei mezzi materiali. Per studiare i fenomeni elettromagnetici tempo-varianti bisogna ricorrere quindi alle equazioni di Maxwell e in particolare è necessario considerare l'accoppiamento che esiste tra i due campi.

Le equazioni in forma differenziale sono:

i. Legge di Gauss	$\nabla \cdot \mathbf{D} = \rho_v$
ii. Legge di Faraday	$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$
iii. Legge del magnetismo di Gauss	$\nabla \cdot \mathbf{B} = 0$
iv. Legge di Ampère-Maxwell	$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}$

Dove:

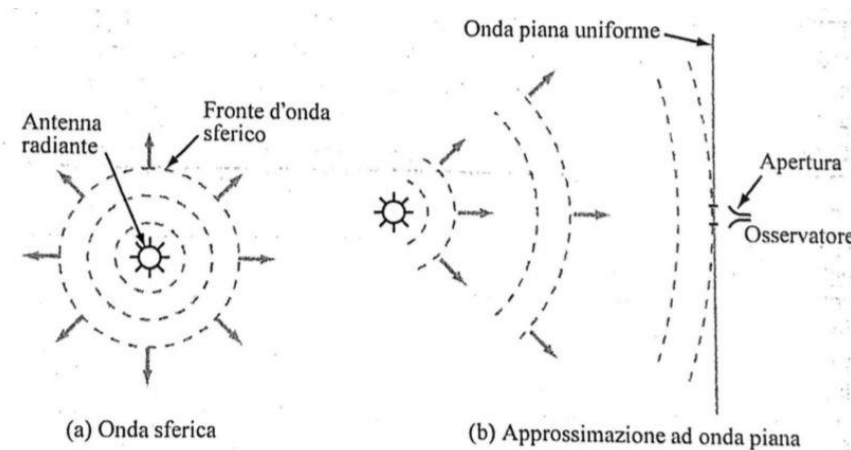
- $\mathbf{D} = \epsilon_0 \epsilon_r \mathbf{E}$ è il vettore induzione elettrica [C/m^2];
- $\mathbf{B} = \mu_0 \mu_r \mathbf{H}$ è il vettore induzione magnetica [Wb/m];
- \mathbf{E} è il vettore campo elettrico [V/m];
- \mathbf{H} è il vettore campo magnetico [A/m];
- \mathbf{J} è la densità superficiale di corrente elettrica [A/m^2];
- ρ_v è la densità volumica di carica elettrica [C/m^3];

e μ_0 permeabilità magnetica del vuoto, μ_r permeabilità magnetica relativa del mezzo, ϵ_0 costante dielettrica del vuoto e ϵ_r costante dielettrica relativa del mezzo.

Nel caso statico invece, dove $\frac{\partial}{\partial t} = 0$, nessuna delle grandezze che compaiono è funzione del tempo e ciò accade quando tutte le cariche sono fisse nello spazio o si muovono ad una velocità costante. Sotto queste ipotesi il campo elettrico e quello magnetico non sono più legati uno all'altro e le quattro equazioni si separano in due sistemi di equazioni, uno che riguarda i campi elettrici \mathbf{E} e \mathbf{D} mentre il secondo i campi magnetici \mathbf{B} e \mathbf{H} . Ciò consente di studiare i fenomeni elettrici e magnetici come casi distinti (elettrostatica e magnetostatica) fino a quando la distribuzione di carica e il flusso rimangono costanti nel tempo.

1.2 Onde piane

Come visto con le leggi di Maxwell, un campo elettrico variabile genera un campo magnetico variabile e viceversa, questo comportamento ciclico genera delle onde elettromagnetiche capaci di propagarsi in qualunque mezzo, compreso il vuoto. Le onde irradiate da una sorgente elettromagnetica, presentano fronti d'onda sferici: nonostante ciò, a grande distanza dal punto di emissione dell'onda, il fronte d'onda appare approssimativamente piano. Le onde piane sono quindi onde elettromagnetiche che si propagano lungo una direzione e rimangono inalterate nelle altre due (non subiscono variazioni nelle direzioni perpendicolari a quella di propagazione), i fronti d'onda sono inoltre infiniti piani paralleli tra loro.



Queste onde sono ovviamente un'astrazione e semplificazione matematica che non corrisponde ad alcun fenomeno della realtà ma, tuttavia sono utili ad approssimare il comportamento dei campi quando la sorgente si trova a grande distanza.

Maxwell analizzando le sue equazioni, dimostrò inoltre che i campi elettrico e magnetico soddisfano l'equazione delle onde nota anche come Equazione di D'Alembert, un'equazione differenziale alle derivate parziali che descrive e sintetizza tutti i fenomeni ondulatori:

$$\nabla^2 \cdot \mathbf{E} = \varepsilon\mu \frac{\partial^2 \mathbf{E}}{\partial t^2}$$

Equazione d'onda per il campo elettrico

$$\nabla^2 \cdot \mathbf{H} = \varepsilon\mu \frac{\partial^2 \mathbf{H}}{\partial t^2}$$

Equazione d'onda per il campo magnetico

Con un riferimento cartesiano (x, y, z) , supponendo di aver a che fare con un'onda piana ed esaminando per semplicità l'equazione relativa al campo elettrico, nell'ipotesi in cui l'onda si propaghi lungo l'asse z e che il campo elettrico abbia componente solo lungo x (onda piana uniforme), l'equazione diventa:

$$\frac{\partial^2 \mathbf{E}_x}{\partial z^2} = \varepsilon\mu \frac{\partial^2 \mathbf{E}_x}{\partial t^2}$$

L'equazione del campo elettrico può essere scritta anche nel dominio della frequenza e assume questa forma:

$$\frac{\partial^2 \mathbf{E}_x}{\partial z^2} = -\omega^2 \varepsilon\mu \mathbf{E}_x$$

è un'equazione differenziale di secondo grado, omogenea, a coefficienti costanti la cui soluzione generale ha la forma:

$$\mathbf{E}_x(z) = E_x^+ e^{-jkz} + E_x^- e^{jkz}$$

Dove:

- $v = \frac{1}{\sqrt{\varepsilon\mu}}$ $\left[\frac{m}{s}\right]$ è la velocità di propagazione dell'onda
- $k = \frac{\omega}{v} = \omega\sqrt{\varepsilon\mu}$ $\left[\frac{1}{m}\right]$ è la costante di propagazione dell'onda

Il primo termine dell'equazione, contenente l'esponentiale negativo, rappresenta un'onda di ampiezza E_x^+ che viaggia nella direzione delle z positive. Il secondo termine invece rappresenta un'onda di ampiezza E_x^- che viaggia nella direzione delle z negative.

Analogamente si può determinare l'espressione del campo magnetico dell'onda piana uniforme, a partire dalla conoscenza del campo elettrico tramite l'equazione di Maxwell del rotore :

$$H_x(z) = H_y(z) = 0$$

$$\mathbf{H}_y(z) = \frac{j}{\omega\mu} \frac{\partial E_x}{\partial z} = \frac{1}{\eta} (E_x^+ e^{-jkz} + E_x^- e^{jkz})$$

Dove :

- $\eta = \frac{\omega\mu}{k} = \sqrt{\frac{\mu}{\varepsilon}}$ $[\Omega]$ è l'impedenza intrinseca del mezzo

1.3 Propagazione onde piane in mezzi con perdite

Nel caso in cui l'onda si propaghi in un mezzo conduttore con conducibilità σ , si definisce la costante di propagazione γ come:

$$\gamma^2 = -\omega^2 \mu (\varepsilon' - j\varepsilon'')$$

Dove

- $\varepsilon' = \varepsilon$
- $\varepsilon'' = \sigma/\omega$

Poiché γ è complessa, si può esprimere come $\gamma = \alpha + j\beta$, con α costante di attenuazione del mezzo e β costante di fase. Queste due costanti sono esprimibili come:

$$\alpha = \omega \left\{ \frac{\mu\varepsilon'}{2} \left[\sqrt{1 + \left(\frac{\varepsilon''}{\varepsilon'}\right)^2} - 1 \right] \right\}^{\frac{1}{2}} \quad [\text{Np/m}] \quad (1.1)$$

$$\beta = \omega \left\{ \frac{\mu\varepsilon'}{2} \left[\sqrt{1 + \left(\frac{\varepsilon''}{\varepsilon'}\right)^2} + 1 \right] \right\}^{\frac{1}{2}} \quad [\text{rad/m}] \quad (1.2)$$

Rimanendo nell'ipotesi in cui il campo elettrico ha come unica componente E_x , l'equazione delle onde assume la forma

$$\frac{\partial^2 E_x}{\partial z^2} = \gamma^2 E_x$$

ed ammette come soluzioni

$$E_x(z) = E_x^+ e^{-\gamma z} + E_x^- e^{\gamma z}$$

Mentre il campo magnetico è

$$H_y(z) = \frac{1}{\eta} (E_x^+ e^{-\gamma z} + E_x^- e^{\gamma z})$$

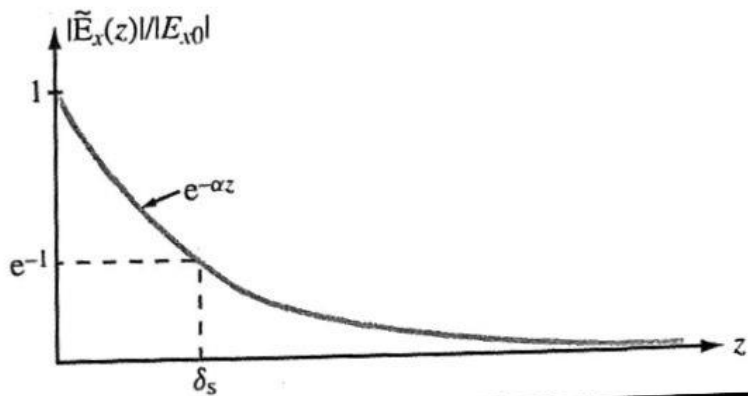
Dove $\eta = \frac{j\omega\mu}{\gamma}$ rappresenta l'impedenza intrinseca del mezzo ed è una variabile complessa.

Dopo una certa distanza $z = \delta_s$ tale che

$$\delta_s = \frac{1}{\alpha}$$

l'ampiezza dell'onda decrescerà di un fattore e^{-1} rispetto al suo valore in $z = 0$. Questa distanza δ_s è chiamata *spessore di penetrazione* del mezzo ed indica proprio quanto un'onda elettromagnetica può penetrare in un mezzo conduttore.

L'ampiezza dei campi decresce esponenzialmente al crescere della distanza e si riduce a e^{-1} quando $z = \delta_s$.



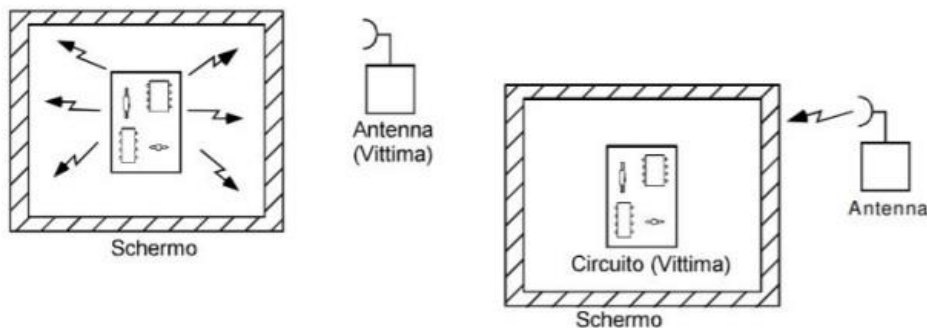
In un dielettrico perfetto si ha $\sigma = 0$ per cui anche $\alpha = 0$ e $\delta_s = \infty$, così nel vuoto un'onda piana può propagarsi senza perdite. Al contrario, se il mezzo è un conduttore con $\sigma = \infty$, l'utilizzo della ε'' nella (1.1) porta ad $\alpha = \infty$ e quindi $\delta_s = 0$.

Le espressioni (1.1) e (1.2), se il mezzo è un dielettrico perfetto in cui $\sigma = 0$, si riducono al caso senza perdite. Per un mezzo con perdite invece, appare in tutte le espressioni il rapporto $\frac{\varepsilon''}{\varepsilon'}$ che gioca un ruolo importante per stabilire l'entità delle perdite.

Se $\frac{\varepsilon''}{\varepsilon'} \ll 1$ il mezzo è detto *dielettrico a basse perdite*, se invece $\frac{\varepsilon''}{\varepsilon'} \gg 1$ il mezzo è definito come un *buon conduttore*.

1.4 Schermatura Elettromagnetica

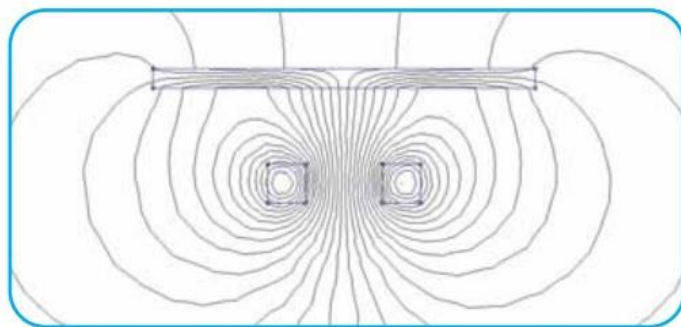
Per schermatura elettromagnetica si intende un sistema in grado di ridurre in una zona dello spazio il campo elettromagnetico che viene generato da una sorgente. Uno *schermo elettromagnetico* costituisce proprio una barriera/contenitore che racchiude completamente o in parte una apparecchiatura elettronica per limitare l'esposizione elettromagnetica. Uno schermo metallico svolge proprio la duplice funzione di schermare l'apparecchio che racchiude dalle interferenze esterne che potrebbero colpirlo e ad evitare che l'apparato emetta dei disturbi rendendolo fuori norma.



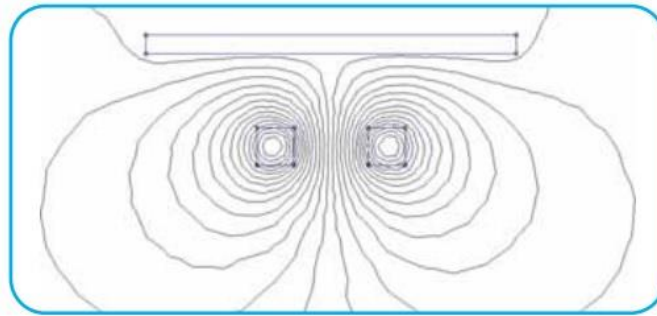
I principali materiali che vengono utilizzati per la schermatura di campi elettromagnetici sono :

- Materiali ad alta permeabilità magnetica
- Materiali ad alta conducibilità elettrica

Il materiale ad alta permeabilità magnetica permette l'abbattimento dell'induzione magnetica mediante l'assorbimento del campo magnetico presente. Il suo comportamento schermante simile ad un "ombrello" di protezione dal campo magnetico che può essere molto intenso vicino allo schermo, ma tende a decadere allontanandosi dal medesimo.



Lo strato di materiale ad elevata conducibilità elettrica in presenza di un campo magnetico variabile (campo induttore) diventa sede di correnti di circolazione, le quali generano a loro volta un campo magnetico di reazione (campo indotto). L'effetto combinato dei campi, indotto e induttore, si traduce in un abbattimento complessivo del campo magnetico totale.



La combinazione dei due materiali, ferromagnetico e conduttivo, permette di realizzare uno schermo con ottime capacità schermanti sia vicino allo schermo, grazie principalmente allo schermo ferromagnetico, sia lontano dallo schermo, grazie allo schermo conduttivo.

Tra i vari materiali che possono essere utilizzati per la realizzazione di uno schermo elettromagnetico troviamo l'alluminio, l'acciaio inossidabile e una lega composta da entrambi. L'utilizzo di alluminio ed acciaio è più efficace per la schermatura a frequenze elevate, soprattutto nei confronti del campo elettrico, anche per piccoli spessori. A frequenze basse, invece, è maggiormente indicato l'acciaio, grazie al suo elevato valore di permeabilità magnetica. In ogni caso un problema rilevante è quello della corrosione che comporta la formazione di uno strato di ossido, la cui conducibilità è nettamente inferiore a quella del metallo. Per ovviare a ciò si possono proteggere le superfici mediante delle vernici protettive o, nel caso dell'alluminio, con la tecnica di anodizzazione.

1.5 Efficacia di schermatura

La bontà di uno schermo elettromagnetico viene valutata numericamente con la cosiddetta *efficacia di schermatura* (Shielding Effectiveness), definita come il rapporto tra l'ampiezza del campo elettrico che si avrebbe in assenza dello schermo e l'ampiezza del campo elettrico che viene trasmesso attraverso lo schermo. Una definizione analoga può essere data per i corrispondenti campi magnetici.

$$SE_{dB} = 20 \log_{10} \left(\left| \frac{E^i}{E^t} \right| \right)$$

$$SE_{dB} = 20 \log_{10} \left(\left| \frac{H^i}{H^t} \right| \right)$$

Prendendo come riferimento l'esempio in figura, è possibile spiegare il principio di funzionamento di uno schermo:

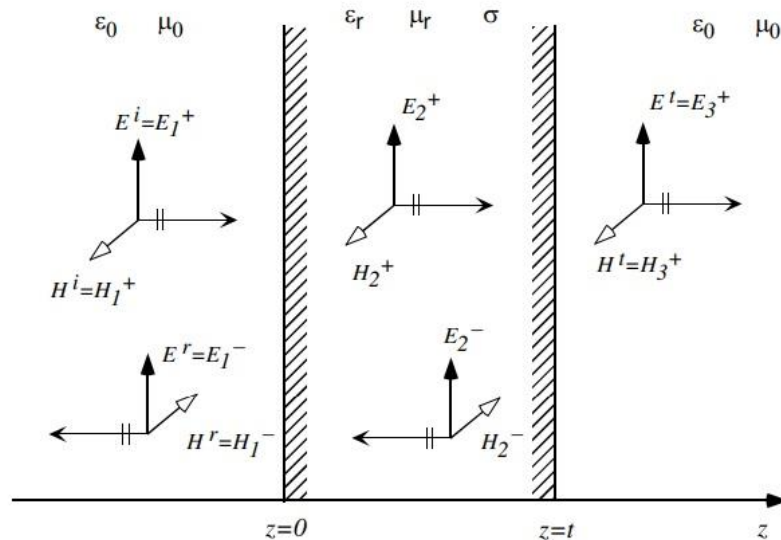


Fig. 5.2 Scomposizione dell'onda incidente su uno schermo metallico.

Nel primo mezzo (aria, con caratteristiche dielettriche ϵ_0 e μ_0) è presente un campo incidente che viene in parte riflesso per la presenza dello schermo (mezzo 2 con caratteristiche elettriche ϵ_r , μ_r e σ) ed in parte trasmesso attraverso lo schermo. Nel secondo mezzo ci sarà quindi la presenza di un'onda diretta ed una riflessa mentre nel terzo mezzo (aria) ci sarà solo l'onda

diretta. Si verificano quindi tre fenomeni che contribuiscono alla diminuzione del campo incidente nel corso dell'attraversamento della barriera:

- *Riflessione* che il campo incidente subisce colpendo la superficie esterna della lastra metallica (mezzo 2). Si presenta alla prima discontinuità aria-metallo.
- *Assorbimento* del campo che è penetrato attraverso lo schermo, che dipende dalle caratteristiche del metallo di cui è costituito lo schermo.
- *Riflessioni multiple* all'interno dello schermo stesso.

Quindi l'espressione relativa all'efficacia di schermatura può essere riscritta in relazione a tali fenomeni come:

$$SE_{dB} = R_{dB} + M_{dB} + A_{dB}$$

dove R indica la perdita dovuta alla riflessione, A quella dovuta all'assorbimento e M è il contributo dovuto alle riflessioni multiple. Si osserva che quest'ultime producono campi che si sommano al campo iniziale trasmesso per cui riducono l'efficacia: il fattore di perdita per le riflessioni multiple M è un numero negativo, mentre R ed A sono positivi.

Supponendo che la sorgente che emette l'onda sia lontana, sappiamo che l'onda incidente può essere considerata localmente piana. Per cui, ricordando le caratteristiche dell'onda definite nella sezione (1.2) e (1.3), i campi elettrico e magnetico possono essere espressi come:

- Nel mezzo 1 (aria):

$$\mathbf{E}_1(z) = E_1^+ e^{-jk_0 z} \mathbf{x} + E_1^- e^{jk_0 z} \mathbf{x} \quad \mathbf{H}_1(z) = \frac{E_1^+}{\eta_0} e^{-jk_0 z} \mathbf{y} + \frac{E_1^-}{\eta_0} e^{jk_0 z} \mathbf{y}$$

- Nel mezzo 2 (metallo con spessore t):

$$\mathbf{E}_2(z) = E_2^+ e^{-j\gamma_2 z} \mathbf{x} + E_2^- e^{j\gamma_2 z} \mathbf{x} \quad \mathbf{H}_2(z) = \frac{E_2^+}{\eta_2} e^{-j\gamma_2 z} \mathbf{y} + \frac{E_2^-}{\eta_2} e^{j\gamma_2 z} \mathbf{y}$$

- Nel mezzo 3 (aria) si ha solo campo trasmesso per cui:

$$\mathbf{E}_3(z) = E_3^+ e^{-jk_0 z} \mathbf{x} \quad \mathbf{H}_3(z) = \frac{E_3^+}{\eta_0} e^{-jk_0 z} \mathbf{y}$$

Imponendo le condizioni di continuità all'interfaccia, e cioè che le componenti tangenziali del campo elettrico e del campo magnetico siano uguali si ottiene:

$$E_1|_{z=0} = E_2|_{z=0} \quad \text{da cui} \quad E_1^+ + E_1^- = E_2^+ + E_2^-$$

$$H_1|_{z=0} = H_2|_{z=0} \quad \text{da cui} \quad \frac{E_1^+}{\eta_0} - \frac{E_1^-}{\eta_0} = \frac{E_2^+}{\eta_2} - \frac{E_2^-}{\eta_2}$$

$$E_2|_{z=t} = E_3|_{z=t} \quad \text{da cui} \quad E_2^+ e^{-j\gamma_2 t} + E_2^- e^{j\gamma_2 t} = E_3^+ e^{-jk_0 t}$$

$$H_2|_{z=t} = H_3|_{z=t} \quad \text{da cui} \quad \frac{E_2^+}{\eta_2} e^{-j\gamma_2 t} - \frac{E_2^-}{\eta_2} e^{j\gamma_2 t} = \frac{E_3^+}{\eta_0} e^{-jk_0 t}$$

Omettendo i passaggi analitici, si raggiunge la seguente espressione che esprime il rapporto tra il campo incidente e quello trasmesso:

$$\frac{E^i}{E^t} = \frac{E_1^+}{E_3^+} = \frac{(\eta_0 + \eta_2)^2}{4\eta_0\eta_2} \left[1 - \left(\frac{\eta_0 - \eta_2}{\eta_0 + \eta_2} \right)^2 e^{-2t/\delta} e^{-2jt/\delta} \right] e^{t/\delta} e^{jt/\delta} e^{-jk_0 t}$$

Quindi la SE in dB diventa:

$$SE_{dB} = 20 \log_{10} \left(\left| \frac{(\eta_0 + \eta_2)^2}{4\eta_0\eta_2} \right| \right) + 20 \log_{10} \left(\left| 1 - \left(\frac{\eta_0 - \eta_2}{\eta_0 + \eta_2} \right)^2 e^{-2t/\delta} e^{-2jt/\delta} \right| \right) \\ + 20 \log_{10} (|e^{t/\delta} e^{jt/\delta} e^{-jk_0 t}|)$$

Dove il primo termine è relativo al contributo di riflessione R_{dB} , il secondo termine alle riflessioni multiple M_{dB} mentre l'ultimo termine al contributo di assorbimento A_{dB} . Inoltre si può notare che se lo schermo è un buon conduttore ($\eta_0 \gg \eta_2$) e $t \gg \delta$ (dove δ è la profondità di penetrazione dell'onda nel metallo) allora M_{dB} può essere trascurato, infatti:

- Se $t \gg \delta$ allora $M_{dB} \cong 0 \rightarrow$ fattore unitario nell'argomento del logaritmo
- Se $t \ll \delta$ allora $M_{dB} < 0 \rightarrow$ le riflessioni multiple riducono la SE dello schermo

1.6 Analisi in frequenza della SE

Nel caso specifico in cui si può considerare $M_{dB} \cong 0$, l'efficacia di schermatura può essere considerata come somma di soli due termini, quello di riflessione e quello di assorbimento:

$$SE_{dB} = R_{dB} + A_{dB} = 20 \log_{10} \left(\left| \frac{\eta_0}{4\eta_2} \right| \right) + 20 \log_{10} (|e^{t/\delta}|)$$

Si cerca ora di elaborare i due termini di questa equazione in modo da esplicitare la loro dipendenza dalla frequenza.

Considerando che $\eta_0 = \sqrt{\mu_0/\varepsilon_0}$, $|\eta_2| = \sqrt{\omega\mu_2/\sigma}$, è possibile riscrivere la *perdita per riflessione* come segue:

$$R_{dB} = 20 \log_{10} \left(\left| \frac{\eta_0}{4\eta_2} \right| \right) = 20 \log_{10} \left(\frac{1}{4} \sqrt{\frac{\mu_0}{\varepsilon_0}} \sqrt{\frac{\sigma}{\omega\mu_2}} \right) = 20 \log_{10} \left(\frac{1}{4} \sqrt{\frac{\sigma}{\omega\mu_R\varepsilon_0}} \right)$$

con μ_R *permeabilità magnetica relativa*. Esplicitando la pulsazione come $\omega = 2\pi f$ e riferendo inoltre la conducibilità del materiale a quella del rame $\sigma_{Cu} = 5.8 \cdot 10^7$ [S/m], ossia imponendo:

$$\sigma = \sigma_{Cu}\sigma_R$$

Allora la perdita per riflessione, in funzione della frequenza, diventa:

$$\begin{aligned} R_{dB} &= 20 \log_{10} \left(\frac{1}{4} \sqrt{\frac{\sigma_{Cu}\sigma_R}{2\pi f\mu_R\varepsilon_0}} \right) = 20 \log_{10} \left(\frac{1}{4} \sqrt{\frac{\sigma_{Cu}}{2\pi\varepsilon_0}} \right) + 20 \log_{10} \left(\frac{\sigma_R}{\mu_R f} \right) \\ &= 168 + 10 \log_{10} \left(\frac{\sigma_R}{\mu_R f} \right) \end{aligned}$$

Quindi si può osservare che R_{dB} è maggiore a bassa frequenza e per materiali ad elevata conducibilità. Inoltre, all'aumentare della frequenza l'efficacia di schermatura decresce di

-10 dB/dec (se rappresentata su un diagramma di Bode) a causa del fenomeno di riflessione. Per quanto riguarda invece le *perdite per assorbimento* invece, si ha che lo spessore di penetrazione δ , i cui parametri sono sempre relativi al rame, è dato dall'espressione

$$\delta = \frac{1}{\sqrt{\pi f \mu \sigma}} = \frac{0,06609}{\sqrt{f \mu_R \sigma_R}}$$

Da cui si ottiene quindi

$$A_{dB} = 20 \log_{10} e^{t/\delta} = 20 \frac{t}{\delta} \log_{10} e = 8,6859 \frac{t}{\delta} = 131,4 t \sqrt{f \mu_R \sigma_R}$$

Si ha quindi un aumento delle perdite per assorbimento all'aumentare del termine sotto radice, quindi della frequenza.

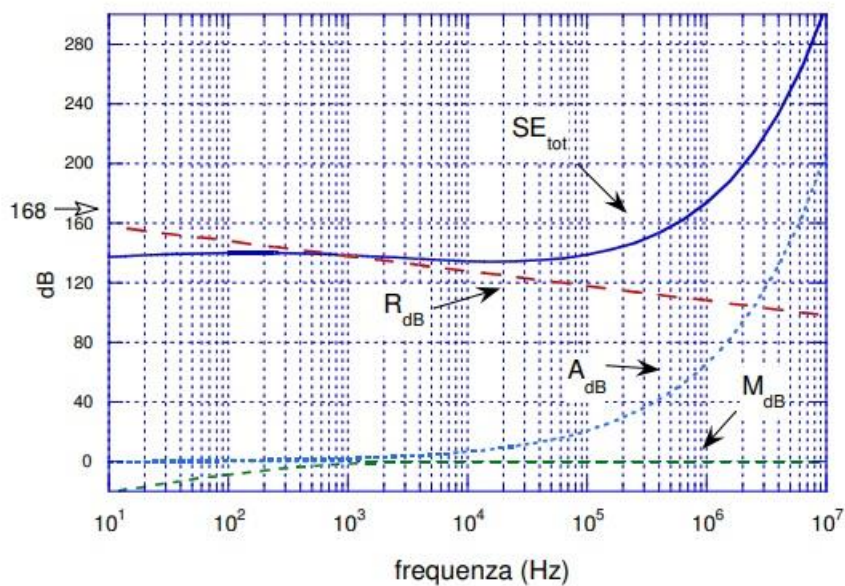


Fig. 5.6 Schermo di rame con spessore $t = 0,5$ mm.

Riportando su un diagramma di Bode l'andamento dei due termini, si può osservare come alle basse frequenze prevalgono le perdite per riflessione mentre alle alte frequenze prevalgono quelle per assorbimento. In particolare, le perdite per riflessione sono consistenti sia per materiali ferrosi che non ferrosi. Ad alta frequenza, invece, la perdita per assorbimento dei materiali ferrosi risulta essere maggiore e quindi l'efficacia di schermatura migliore.

CAPITOLO 2 – SE per un vetro AGC

2.1 Analisi del vetro

AGC è l'acronimo di *Asahi Glass Company* ed è l'azienda leader a livello mondiale per la produzione di vetro piano: è specializzata nella produzione e nella lavorazione di vetro piano per l'industria edilizia (vetrate esterne e vetri decorativi per interni), per quella automobilistica, per le applicazioni solari e per altre industrie specializzate.

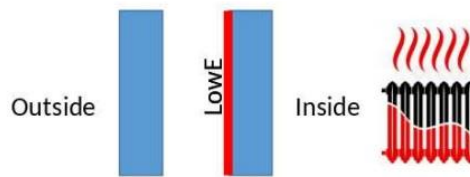
La valutazione dell'efficacia di schermatura è stata studiata attraverso diversi test sperimentali effettuati su un campione di vetro composito soggetto ad onde piane.

Il campione utilizzato è un vetro basso emissivo, ovvero un vetro a isolamento termico che permette di limitare la dispersione termica all'interno degli edifici: secondo l'azienda AGC, infatti, in media nelle abitazioni un quarto di calore viene disperso attraverso i muri e fino ad un 40% tramite le finestre.

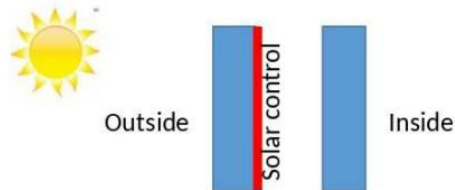
Questo tipo di vetro è progettato infatti per prevenire la dissipazione di calore verso l'ambiente esterno (o viceversa a causa della radiazione solare): per ottenere questa proprietà, sui vetri in questione viene applicato un rivestimento termico costituito da strati di ossido di metallo (*coating*) che riduce drasticamente il trasferimento di calore. Questo rivestimento è trasparente alle bande elettromagnetiche del visibile, ma è in grado di controllare le onde infrarosse che entrano o escono da una stanza.

Si possono individuare due tipi di rivestimento:

Low E - keep heat inside



Solar control -avoid overhear inside building

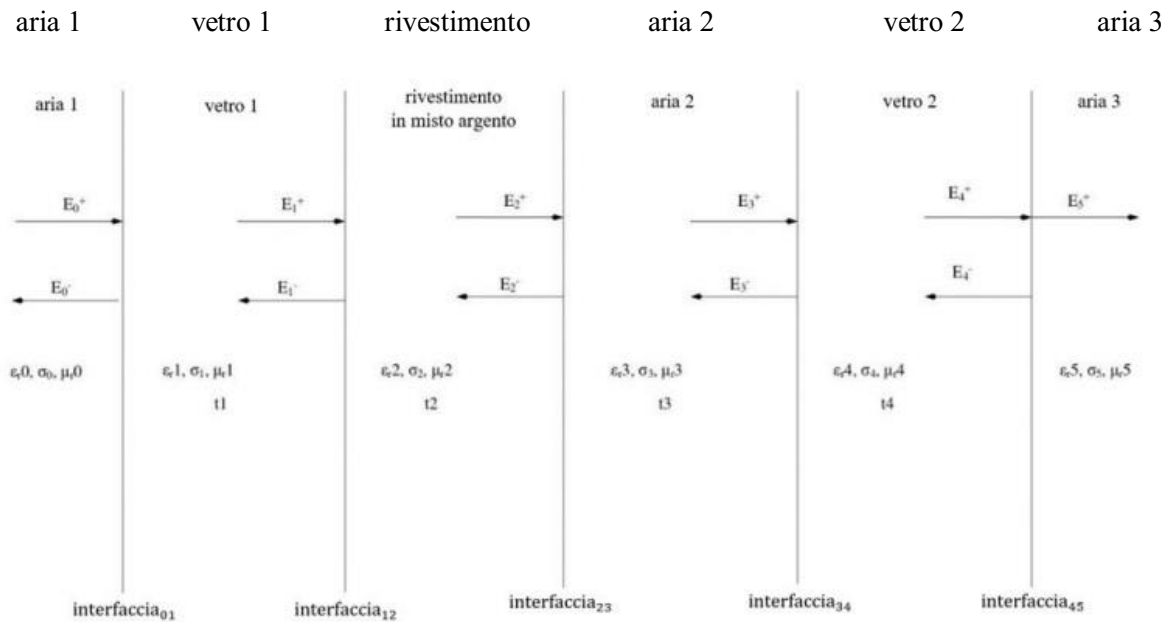


- *Rivestimenti Low-E* che sono stati progettati per trattenere il calore all'interno degli edifici in inverno e vengono posti sul lato interno del doppio vetro (*DGU*).
- *Rivestimenti solar control* (a controllo solare) sono stati progettati invece per limitare la quantità di calore solare che li attraversa in modo da mantenere l'edificio più fresco in estate, e per ridurre quindi il consumo di energia legato alla climatizzazione. Questo tipo di rivestimento viene posto sul lato esterno del doppio vetro.

Tuttavia, l'utilizzo del doppio vetro (DGU) con rivestimento impedisce al segnale di entrare negli edifici, in particolare blocca la penetrazione delle onde radio al loro interno proprio perché composto da un rivestimento a base di metallo per l'isolamento termico. Una doppia vetrata con rivestimento attenua infatti il segnale di 30 dB, il che significa che solo lo 0,1% del segnale riesce a penetrare all'interno dell'edificio, mentre un vetro senza rivestimento riesce a far penetrare il 50% del segnale attenuandolo solo di 3 dB.

2.2 Descrizione del modello

La vetrata (unità a doppio vetro) può essere assimilata ad uno schermo elettromagnetico multistrato come mostrato in figura:



Il primo strato è rappresentato dall'aria esterna all'edificio da cui parte un'onda elettromagnetica che incide normalmente, il segnale poi andrà ad incidere sui successivi strati che sono nell'ordine: la lastra di vetro, il rivestimento, lo strato di aria interposto tra i due vetri, la seconda lastra di vetro e infine l'ultimo strato è costituito dall'aria presente all'interno dell'edificio. Non essendoci un'interfaccia successiva a quella tra *vetro 2* e *aria 3*, nell'ultimo mezzo non si ha un'onda riflessa ma solo un'onda trasmessa.

L'efficacia di schermatura SE del multistrato può essere calcolata come:

$$SE_{dB} = 20 \log_{10} \left(\left| \frac{E_0^+}{E_5^+} \right| \right)$$

Dove E_0^+ è l'ampiezza del campo elettrico incidente sull'unità a doppio vetro mentre E_5^+ è l'ampiezza del campo che si propaga nell'aria (interno dell'edificio).

Di seguito vengono riportati i valori delle ampiezze dei campi presenti negli strati intermedi:

$$\bullet \quad E_0^+ = \frac{1}{\tau_{01}} (E_1^+ e^{j\gamma_1 t_1} + \Gamma_{01} E_1^- e^{-j\gamma_1 t_1}) \quad E_0^- = \frac{1}{\tau_{01}} (\Gamma_{01} E_1^+ e^{j\gamma_1 t_1} + E_1^- e^{-j\gamma_1 t_1})$$

Dove $\Gamma_{01} = \frac{\eta_1 - \eta_0}{\eta_1 + \eta_0}$ è il coefficiente di riflessione e $\tau_{01} = 1 + \Gamma$ il coefficiente di trasmissione.

$$\bullet \quad E_1^+ = \frac{1}{\tau_{12}} (E_2^+ e^{j\gamma_2 t_2} + \Gamma_{12} E_2^- e^{-j\gamma_2 t_2}) \quad E_1^- = \frac{1}{\tau_{12}} (\Gamma_{12} E_2^+ e^{j\gamma_2 t_2} + E_2^- e^{-j\gamma_2 t_2})$$

$$\bullet \quad E_2^+ = \frac{1}{\tau_{23}} (E_3^+ e^{j\gamma_3 t_3} + \Gamma_{23} E_3^- e^{-j\gamma_3 t_3}) \quad E_2^- = \frac{1}{\tau_{23}} (\Gamma_{23} E_3^+ e^{j\gamma_3 t_3} + E_3^- e^{-j\gamma_3 t_3})$$

$$\bullet \quad E_3^+ = \frac{1}{\tau_{34}} (E_4^+ e^{j\gamma_4 t_4} + \Gamma_{34} E_4^- e^{-j\gamma_4 t_4}) \quad E_3^- = \frac{1}{\tau_{34}} (\Gamma_{34} E_4^+ e^{j\gamma_4 t_4} + E_4^- e^{-j\gamma_4 t_4})$$

$$\bullet \quad E_4^+ = \frac{1}{\tau_{45}} (E_5^+ e^{j\gamma_5 t_5}) \quad E_4^- = \frac{1}{\tau_{45}} (\Gamma_{45} E_5^+ e^{j\gamma_5 t_5})$$

A partire da queste relazioni e assumendo che $E_5^+ = 1 \text{ V/m}$, si può ricavare l'ampiezza del campo incidente sull'unità a doppio vetro e quindi calcolare la SE:

$$E_0^+ = \frac{1}{\tau_{45} \tau_{34} \tau_{23} \tau_{12} \tau_{01}} E_5^+$$

2.3 Introduzione a Python

Python è un linguaggio di programmazione di “alto livello” dalla sintassi semplice, ricco di librerie, ideato nei primi anni novanta dall'informatico olandese Guido Van Rossum. Diversi sono gli ambiti di applicazione di questo linguaggio, come lo sviluppo di siti o applicazioni Web e desktop, realizzazione di interfacce grafiche, amministrazione di sistema, database, giochi, grafica 3D, eccetera.

Ad oggi esistono numerosi linguaggi di programmazione ma ciò che spinge gli utenti ad utilizzare Python sono proprio le sue caratteristiche:

- È completamente gratuito ed è possibile usarlo e distribuirlo senza restrizioni di copyright.
- È un linguaggio multi-paradigma: supporta sia la programmazione procedurale (che fa uso delle funzioni), sia la programmazione ad oggetti e supporta anche diversi elementi

della programmazione funzionale (come iteratori e generatori).

- È un linguaggio portabile, è possibile usarlo su diverse piattaforme come: Unix, Linux, Windows, DOS, cellulari Android e iOS. Ciò è possibile perché si tratta di un linguaggio interpretato (lo stesso codice può essere eseguito su qualsiasi piattaforma purché abbia l'interprete Python installato).
- È un linguaggio che adotta un meccanismo di *garbage collection* che si occupa automaticamente dell'allocazione e del rilascio della memoria. Questo consente al programmatore di usare variabili liberamente, senza dover necessariamente dichiararle e allocare spazi di memoria manualmente (cosa invece necessaria in linguaggi di più basso livello come il C o il C++).

Non esiste soltanto una versione di Python ed è dunque importante saper scegliere quella giusta per il proprio scopo. Le versioni di si dividono principalmente in *Python 2* e *Python 3*: quest'ultima è la versione più moderna e supportata del linguaggio, con dei cambiamenti notevoli che la rendono incompatibile con Python 2.

Per realizzare un programma in Python, è necessario installare sul computer, il relativo ambiente di sviluppo, che contiene tutto il necessario per l'esecuzione del codice sorgente: nel caso di questa simulazione per il calcolo dell'SE si è scelto di installare il software *JupyterLab*. Inoltre, per la scrittura del codice che verrà riportato in seguito, sono state installate e utilizzate tre librerie, ovvero delle raccolte di funzioni e programmi che vengono richiamate dai programmi informatici per aggiungere nuove funzionalità al codice, che sono:

- *NumPy* è una libreria fondamentale su cui si basano la maggior parte delle librerie di elaborazione dati Python, aggiunge supporto a grandi matrici e array multidimensionali insieme a una vasta collezione di funzioni matematiche di alto livello per poter operare efficientemente su queste strutture dati.
- *SciPy*, una collezione di algoritmi matematici ed altre funzioni particolarmente utilizzate in ambito scientifico. SciPy è costruita sulla base di NumPy.
- *Matplotlib* che è invece la libreria per la creazione di grafici per il linguaggio Python.

2.4 Algoritmo per il calcolo della SE

Per il calcolo dell'efficacia di schermatura dell'unità a doppio vetro descritta in precedenza, in cui un'onda elettromagnetica incide in modo obliquo ad essa, è stato creato un codice in linguaggio Python sulla base di un preesistente codice sviluppato in linguaggio C nelle tesi di Laura Marinaro e Tomasz De Clerico.

Verranno ora riportate delle sottosezioni in cui verrà analizzata e spiegata ogni parte dell'algoritmo.

2.4.1 Importazione delle librerie

Nel linguaggio Python per poter utilizzare funzioni appartenenti ad una libreria esterna è necessario importare tale libreria nel programma tramite l'istruzione *import* seguita dal nome della libreria, in questo caso le librerie utilizzate sono quelle definite nella sezione (2.3). Se l'importazione non dovesse andare a buon fine, vorrebbe dire che la libreria non è ancora stata installata sulla versione Python che sto usando, quindi è necessario prima installarla.

In Python, inoltre, è possibile rinominare la libreria con un alias più breve:

```
import nome_libreria as nome_alias
```

In genere un alias è composto da due o tre lettere che indicano la contrazione del nome della libreria. Ad esempio, "scp" invece di "scipy", "np" invece di "numpy", ecc. Non ci sono convenzioni, ogni sviluppatore può scegliere l'alias che preferisce.

```
# importazione delle librerie utilizzate
import scipy.constants as scp
import numpy as np
import matplotlib.pyplot as plt
```

In seguito, sono poi riportate alcune costanti fisiche (e i loro relativi valori) utilizzate nel codice e provenienti in particolare dalla libreria *SciPy*. Queste costanti sono nell'ordine: pigreco, permittività dielettrica nel vuoto, permeabilità magnetica nel vuoto e la velocità della luce.

```

# Costanti fisiche
pi=scp.pi
eps0=scp.epsilon_0
mu0=scp.mu_0
c=scp.c
print("sciPy - pi = %.16f"%pi)
print("sciPy - eps0 = %.16g"%eps0)
print("sciPy - mu0 = %.16g"%mu0)
print("sciPy - c = %.1f"%c)

sciPy - pi = 3.1415926535897931
sciPy - eps0 = 8.854187812800001e-12
sciPy - mu0 = 1.25663706212e-06
sciPy - c = 299792458.0

```

2.4.2 Funzione per il calcolo dei parametri

Per poter tenere un codice quanto più leggero possibile, uniforme e semplice è molto importante evitare ripetizioni superflue: uno dei concetti fondamentali nel mondo della programmazione è infatti il concetto di riutilizzo del codice. Più un programma diventa complesso e strutturato, più saranno le righe di codice che lo compongono, per questo è possibile creare delle opportune funzioni che una volta definite possono essere utilizzate più volte all'interno del codice, evitando inutili ripetizioni e mantenendo così il software ordinato.

Per definire una funzione in Python, viene utilizzata la parola chiave *def* seguita dal nome che si vuole dare alla funzione (è preferibile utilizzare nomi che siano rappresentativi dello scopo della funzione). Il nome viene seguito poi da una coppia di parentesi, all'interno delle quali vengono inseriti i parametri/variabili che vengono passate alla funzione. Seguiranno poi una serie di istruzioni che implementano la funzione e infine l'istruzione *return* che restituisce al codice i valori trovati con la funzione.

def nome_funzione (variabili passate)

...istruzioni...

return variabili_di_ritorno

Il caso in esame è costituito da un'unità a doppio vetro composta da strati di materiali differenti, per cui è necessario definire una funzione *materiale* che permette di ricavare i parametri elettromagnetici di ogni strato, ovvero la costante di propagazione complessa (*gamma*) e l'impedenza caratteristica del materiale (*eta*).

```

# Calcolo dei parametri di propagazione nel materiale
def materiale (strato):
    epsr=strato[1]
    mur=strato[2]
    sigma=strato[3]
    omega=2*scp.pi*freq
    epsr2=sigma/(omega*scp.epsilon_0)
    tgd2=(epsr2**2)/(epsr**2)
    alfa=omega * np.sqrt (mur * scp.mu_0 * epsr * scp.epsilon_0 * 0.5 * (np.sqrt (1 + tgd2) - 1))
    beta=omega * np.sqrt (mur * scp.mu_0 * epsr * scp.epsilon_0 * 0.5 * (np.sqrt (1 + tgd2) + 1))
    gamma=alfa + beta * 1j
    eta=np.sqrt (mur * scp.mu_0 / ((epsr - epsr2 * 1j) * scp.epsilon_0))
    return gamma, eta

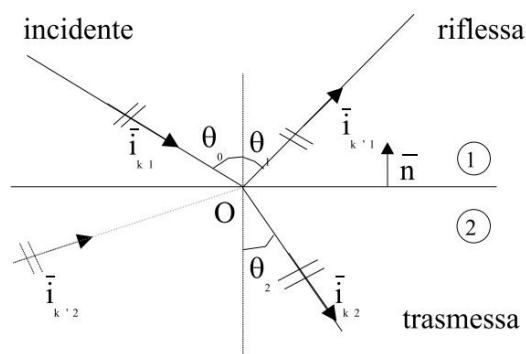
```

La funzione prende in ingresso la variabile *strato*, ovvero una struttura che viene definita poi nel *main* e che per ogni *strato* include cinque campi di valori che nell'ordine sono: nome del materiale, permittività dielettrica relativa, permeabilità magnetica relativa, conducibilità elettrica e spessore dello strato.

All'interno della funzione vengono inizialmente assegnati i parametri di *strato* (con il proprio indice) a delle variabili locali, successivamente viene ricavata la pulsazione *omega* a partire dalla variabile *freq* (anch'essa definita nel *main* e che costituisce un determinato intervallo di frequenze), e la parte immaginaria della permittività elettrica relativa complessa nel mezzo (*epsr2*). Vengono calcolate poi la parte reale (*alfa*) e la parte immaginaria (*beta*) della costante di propagazione complessa. Con i valori finora ricavati è possibile calcolare le variabili *gamma* e *eta* che costituiscono i valori di ritorno della funzione.

2.4.3 Leggi di Snell

Un'onda elettromagnetica che incide su una superficie di separazione tra due mezzi produce un'onda riflessa e un'onda trasmessa. Considerando una superficie che separa due mezzi con indici di rifrazione diversi e un'onda che incide in modo obliquo su questa superficie, l'onda riflessa formerà un angolo di riflessione che corrisponde esattamente all'angolo di incidenza



Dove in questo caso θ_0 è l'angolo che forma l'onda incidente con la normale alla superficie di separazione mentre θ_1 è l'angolo formato dall'onda riflessa con la normale alla superficie. Allora quello che succede nel fenomeno della riflessione è che questi due angoli sono uguali tra loro:

$$\theta_0 = \theta_1$$

Questa legge prende il nome di *Legge di Snell per la riflessione*.

Come già detto in precedenza però, l'onda incidente verrà in parte anche rifratta: se chiamo θ_2 l'angolo formato dall'onda rifratta con la normale alla superficie, questo angolo sarà anch'esso legato all'angolo di incidenza tramite la *Legge di Snell per la rifrazione*:

$$n_1 \sin \theta_0 = n_2 \sin \theta_2$$

Dove n_1 ed n_2 sono gli indici di rifrazione dei due mezzi separati dalla superficie.

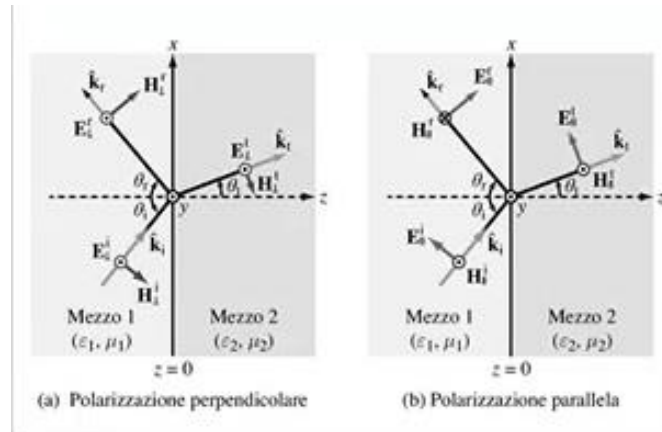
Per cui:

```
# calcolo degli angoli nell'interfaccia tra due mezzi con la legge di Snell
def snell (n1,n2,ti):
    theta_inc=ti
    n1=n1
    n2=n2
    theta_rifl=theta_inc
    arg=np.sin(theta_inc)*n1/n2
    if (arg>1):
        theta_rifr=scp.pi
    else:
        theta_rifr=np.arcsin(arg)
    return theta_rifl, theta_rifr
```

2.4.4 Polarizzazione perpendicolare

Prendendo in considerazione una qualsiasi onda elettromagnetica, il campo elettrico e il campo magnetico dell'onda possono oscillare nello spazio in una qualunque direzione purché si mantengano perpendicolari tra loro e perpendicolari alla direzione di propagazione dell'onda. Un'onda viene detta *polarizzata* se la direzione di oscillazione del campo elettrico è definita nello spazio, di conseguenza lo sarà anche quella del campo magnetico in quanto si manterrà perpendicolare a quella del campo elettrico. Per descrivere lo stato di polarizzazione si fa riferimento per convenzione al solo vettore del campo elettrico. Gli stati di polarizzazione di

un'onda possono essere di tre tipi: lineare, circolare ed ellittica a seconda della direzione di oscillazione del campo elettrico. Un'onda con polarizzazione qualsiasi può essere descritta come la sovrapposizione di due onde polarizzate ortogonalmente l'una rispetto all'altra: una avrà il campo elettrico parallelo al piano d'incidenza (*polarizzazione parallela*) mentre nell'altra il campo elettrico è perpendicolare al piano (*polarizzazione perpendicolare*).



Nel caso in esame discusso in questa tesi, si considera un'onda con incidenza obliqua polarizzata perpendicolarmente mentre il caso di polarizzazione parallela verrà analizzato nella tesi di Eleonora Bianchi.

Come visto nella sezione (2.4.3), se l'onda incide in modo obliquo rispetto all'interfaccia tra i due mezzi, i fenomeni di riflessione e rifrazione sono descritti dalle *Leggi di Snell*. Nel caso di incidenza normale, il coefficiente di riflessione Γ e quello di rifrazione τ all'interfaccia tra i due mezzi sono indipendenti dalla polarizzazione dell'onda incidente. Ciò non accade nel caso di incidenza obliqua, per cui i coefficienti di riflessione e rifrazione nel caso di polarizzazione perpendicolare possono essere scritti come:

$$\Gamma_{\perp} = \frac{E_{\perp 0}^r}{E_{\perp 0}^i} = \frac{\eta_2 \cos \theta_i - \eta_1 \cos \theta_t}{\eta_2 \cos \theta_i + \eta_1 \cos \theta_t}$$

$$\tau_{\perp} = 1 + \Gamma_{\perp}$$

Dove:

- θ_i e θ_t sono gli angoli di incidenza e trasmissione (o rifrazione).

- η_1 e η_2 sono l'impedenza intrinseca dei due mezzi.
- $E_{\perp 0}^r$ e $E_{\perp 0}^i$ sono le componenti perpendicolari del campo elettrico dell'onda riflessa e incidente calcolate all'interfaccia tra i due mezzi.

Nel codice per il calcolo della SE è stata creata una funzione *interf_obl_perp* che ha proprio il compito di calcolare i coefficienti di riflessione (*rho*) e di trasmissione (*tt*) per ogni interfaccia tra i vari strati della vetrata, nel caso di polarizzazione perpendicolare dell'onda:

```
# Calcolo dei coefficienti di trasmissione e riflessione per polarizzazione normale (perpendicolare)
def interf_obl_perp (e1,e2,t1,t2):
    eta1=e1
    eta2=e2
    theta1=t1
    theta2=t2
    rho = (eta2*np.cos(theta1)-eta1*np.cos(theta2))/(eta2*np.cos(theta1)+eta1*np.cos(theta2))
    tt = 1 + rho
    return rho, tt
```

È stata creata una funzione anche per il calcolo degli stessi coefficienti nel caso però di polarizzazione parallela dell'onda, analizzato da Eleonora Bianchi:

```
# Calcolo dei coefficienti di trasmissione e riflessione per polarizzazione parallela
def interf_obl_par (e1,e2,t1,t2):
    eta1=e1
    eta2=e2
    theta1=t1
    theta2=t2
    rho = (eta2*np.cos(theta2)-eta1*np.cos(theta1))/(eta2*np.cos(theta2)+eta1*np.cos(theta1))
    tt = (1 + rho) * np.cos(theta1) / np.cos(theta2)
    return rho, tt
```

2.4.5 Funzione per il calcolo del campo elettrico in ogni strato

La funzione *campi* permette di calcolare il campo elettrico in ogni strato della vetrata a partire dalla conoscenza del campo elettrico nell'ultimo strato:

```

# Calcolo del campo elettrico per l'onda progressiva e regressiva in uno strato
def campi (gamma, thick, rho, tt, e_p, e_m):
    gamma=gamma
    thick=thick
    e_p=e_p
    e_m=e_m
    rho=rho
    tt=tt
    exp_piu = np.exp (gamma * thick)
    exp_meno = np.exp (- gamma * thick)
    e_piu = (exp_piu * e_p + rho * exp_meno * e_m) / tt
    e_meno = (rho * exp_piu * e_p + exp_meno * e_m) / tt
    return e_piu, e_meno

```

I parametri in ingresso alla funzione sono: la costante di propagazione (*gamma*), lo spessore (*thick*), i coefficienti di riflessione e trasmissione (*rho* e *tt*), campo elettrico trasmesso e riflesso (*e_p* e *e_m*). I valori di questi parametri vengono assegnati a delle variabili locali, successivamente vengono calcolati due termini esponenziali, *exp_piu* e *exp_meno*, tramite la funzione *exp* contenuta nella libreria *NumPy*: questi due termini vengono utilizzati per il calcolo del campo elettrico riflesso e trasmesso *e_piu* e *e_meno*, che costituiscono i valori di ritorno della funzione.

2.4.6 Funzione per il calcolo della SE

La funzione *SE* costituisce a tutti gli effetti la funzione principale dell'intero programma, in quanto grazie ad essa è possibile calcolare l'efficacia di schermatura dell'unità a doppio vetro. Tale funzione presenta come parametri in ingresso la frequenza dell'onda che incide sul primo strato (*freq*) e l'angolo *ti* e presenta tra le sue istruzioni una serie di iterazioni in cui vengono richiamate anche le funzioni descritte e analizzate fino ad ora.

```

# Calcolo dell'efficacia di schermatura della struttura multistrato per polarizzazione normale (perpendicolare)
def SE_obl_perp(freq,ti):
    theta_m[0]=ti
    for i in range(nstrati):
        n[i]=np.sqrt(strati[i][1]*strati[i][2])
        gamma[i],eta[i]=materiale(strati[i])

    for i in range(1,nstrati,1):
        theta_rifl,theta_m[i]=snell(n[i-1],n[i],theta_m[i-1])
    for i in range(nstrati-1):
        rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
    i=nstrati-1
    e_p[i] = 1 + 0j;
    e_m[i] = 0 + 0j;
    for i in range(nstrati-2, -1, -1):
        e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])
    SE_lin = np.abs(e_p[0])
    return SE_lin

```

In questa funzione viene ampiamente utilizzato il ciclo *for*, in particolare cicli *for* associati alla funzione *range*: il linguaggio Python quando si vuole lavorare su sequenze di numeri, fornisce una funzione *range* che permette di specificare un valore iniziale o *start* (incluso), un valore finale *stop* (escluso) e un passo di avanzamento (*step*). La sintassi è la seguente:

for in range (start,stop,step):

....

Nel primo ciclo *for*, che esegue un numero di iterazioni pari al numero di strati della vetrata, vengono calcolati gli indici di rifrazione ($n[i]$), la costante di propagazione ($\gamma[i]$) e l'impedenza caratteristica ($\eta[i]$) per ogni strato, richiamando la funzione *materiale*.

In un secondo ciclo *for*, tramite la funzione *snell* vengono poi calcolati gli angoli di riflessione e rifrazione nell'interfaccia tra i due mezzi.

Successivamente è presente un ciclo *for* con un range pari al numero delle interfacce, ovvero $n_{strati}-1$, che tramite la funzione *interf_obl_perp* va a calcolare invece i coefficienti ρ e t (di riflessione e trasmissione) per ogni interfaccia.

L'ultimo ciclo *for* è caratterizzato da $start=n_{strati}-2$, $stop=-1$, $step=-1$ ovvero è un ciclo che parte da $i=n_{strati}-2$ e si conclude quando $i=0$ (viene iterato al contrario). In questo ciclo, attraverso la funzione *campi* vengono calcolati il campo elettrico riflesso e trasmesso in ogni strato a partire dallo strato *vetro 2* fino al primo strato *aria 1*, da cui è possibile poi ricavare l'efficacia di schermatura.

2.4.7 Main (SE prima dell'ottimizzazione)

```
# Calcolo dell'SE prima dell'ottimizzazione
nstrati=6
theta_iniziale=85/180*np.pi
n=[0]*nstrati
gamma=[0]*nstrati
eta=[0]*nstrati
theta_m=[0]*nstrati
theta_m[0]=theta_iniziale
tt=[0]*(nstrati-1)
rho=[0]*(nstrati-1)
e_p=[0]*(nstrati)
e_m=[0]*(nstrati)
#strati = [("aria 1",1,1,0,0),
#          ("vetro 1",7,1,0.001,0.008264),
#          ("metallo",1,1,61730000,6.5e-9),
#          ("aria 2",1,1,0,0.01218),
#          ("vetro 2",7,1,0.001,0.003615),
#          ("aria 3",1,1,0,0)]
strati = [("aria 1",1,1,0,0),
          ("vetro 1",7,1,0.001,x0[0]),
          ("metallo",1,1,61730000,x0[1]),
          ("aria 2",1,1,0,x0[2]),
          ("vetro 2",7,1,0.001,x0[3]),
          ("aria 3",1,1,0,0)]
#freq=1e9
freq = np.linspace ( start = 100e6      # Lower limit (Hz)
                    , stop = 10e9      # upper limit (Hz)
                    , num = 501      # generate 501 points between 0 and 3
                    )
for i in range(nstrati):
    n[i]=np.sqrt(strati[i][1]*strati[i][2])
    gamma[i],eta[i]=materiale(strati[i])

for i in range(1,nstrati,1):
    theta_rifl,theta_m[i]=snell(n[i-1],n[i],theta_m[i-1])
#
# scegliere se calcolare 'interf_obl_perp' o 'interf_obl_par'
#
for i in range(nstrati-1):
    rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
# print("Interfaccia:",i," - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
i=nstrati-1
e_p[i] = 1 + 0j;
e_m[i] = 0 + 0j;
#print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"V/m")
for i in range(nstrati-2, -1, -1):
    e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])
# print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"V/m")
#print("SE (dB) = ", np.round(20 * np.Log10(np.abs(e_p[0])),2))
#fstart = 0.1 # IN GHz
#fstop = 10 # IN GHz
#fstep = 1e-3 # IN GHz
#fnum = (fstop - fstart) / fstep + 1
```

```

#print("x = ",x)
#print("y = ",y)
plt.rc('xtick', labels=12)
plt.rc('ytick', labels=12)
plt.rc('axes', labels=12)
plt.xlabel('Frequency (GHz)')
plt.ylabel('SE (dB)')
y = 20 * np.log10(SE_obl_perp(freq,0/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,15/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,30/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,45/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,60/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,67.5/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,75/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
plt.show()

```

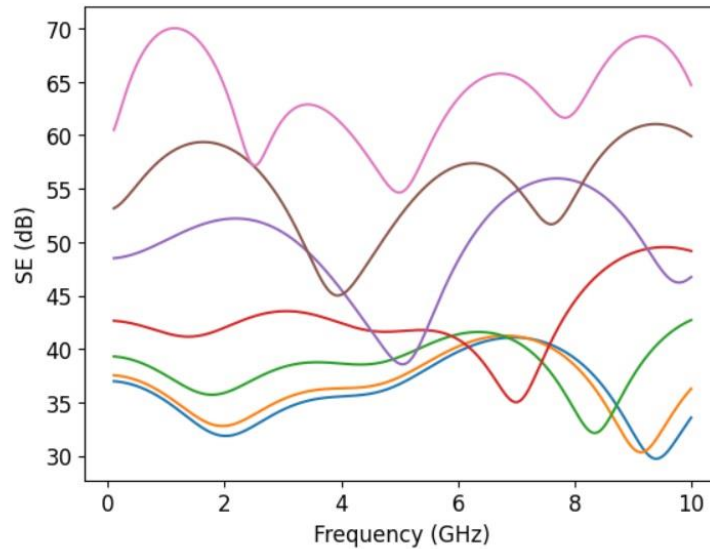
Come mostrato in figura, all'interno del main per il calcolo dell'efficacia di schermatura prima dell'ottimizzazione, viene definita la variabile *nstrati* a cui è assegnato il valore 6 ovvero il numero degli strati che compongono l'unità a doppio vetro presa in esame (aria 1, vetro 1, metallo, aria 2, vetro 2 e aria 3).

Successivamente, i valori dei parametri che vengono calcolati, tramite le varie funzioni definite in precedenza, vengono memorizzati all'interno di vettori di opportuna lunghezza (*nomevettore=[0]*(lunghezza)*).

Viene poi creata una struct *strati* contenente appunto i sei strati della vetrata in esame, e in ciascuno strato viene specificato: il nome del materiale, permittività dielettrica relativa, permeabilità magnetica relativa, conducibilità elettrica e spessore dello strato.

Tramite la funzione *linspace* della libreria Numpy, viene creata la variabile *freq*: tale funzione permette di definire un array composto da una serie di N elementi numerici equidistribuiti tra due estremi, in questo caso gli estremi sono *start=100 MHz*, *stop= 10 GHz* mentre il numero di elementi è *num=501*. Ora per tutti i valori di frequenza e con una variazione dell'angolo di incidenza, viene calcolata l'efficacia di schermatura (riportata in dB) per ciascun angolo scelto tramite la funzione *SE_obl_perp*.

A questo punto vengono riportati in un grafico i valori trovati: tramite i comandi *xlabel* e *ylabel* della libreria *Matplotlib* vengono assegnati i nomi agli assi, i valori della frequenza vengono riportati in GHz sull'asse delle ascisse mentre i valori dell'SE in dB sull'asse delle ordinate.



2.4.8 SE medio

Oltre a calcolare l'SE al variare della frequenza e dell'angolo d'incidenza, è stato anche calcolato l'SE medio nel caso di polarizzazione parallela e polarizzazione perpendicolare.

```
# Grafico dell'SE prima dell'ottimizzazione
freq = np.linspace ( start = 100e6 # Lower limit (Hz)
                    , stop = 10e9 # upper limit (Hz)
                    , num = 501 # generate 501 points between 0 and 3
                    )

for i in range(nstrati):
    n[i]=np.sqrt(strati[i][1]*strati[i][2])
    gamma[i],eta[i]=materiale(strati[i])
    # print("Materiale:",strati[i][0]," - Indice di rifrazione =",np.round(n[i]))
    # print("Materiale:",strati[i][0]," - Costante di propagazione =",np.round(gamma[i],2),"(1/m), Impedenza caratteristica =",np.round(eta[i]

for i in range(1,nstrati,1):
    theta_rifl,theta_m[i]=snell(n[i-1],n[i],theta_m[i-1])
    # print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
    #print("SE (dB) = ", np.round(20 * np.Log10(np.abs(e_p[0])),2))
    #fstart = 0.1 # IN GHz
    #fstop = 10 # IN GHz
    #fstep = 1e-3 # IN GHz
    #fnum = (fstop - fstart) / fstep + 1

#print("x = ",x)
#print("y = ",y)
plt.rc('xtick', labels=12)
plt.rc('ytick', labels=12)
plt.rc('axes', labels=12)
plt.xlabel('Frequency (GHz)')
plt.ylabel('SE medio (dB)')
for i in range(nstrati-1):
    rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
    # print("Interfaccia:",i," - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
i=nstrati-1
e_p[i] = 1 + 0j;
e_m[i] = 0 + 0j;
#print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
```

```

for i in range(nstrati-2, -1, -1):
    e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])

n_angoli = 181
somma_obl_perp = 0
for i in range(0,n_angoli - 1,1):
    somma_obl_perp = somma_obl_perp + 1 / SE_obl_perp(freq, i * 0.5 * np.pi / (n_angoli - 1))
somma_obl_perp = somma_obl_perp / (n_angoli - 1)
y = 20 * np.log10(1 / somma_obl_perp) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y, label='Polarizzazione Normale')
for i in range(nstrati-1):
    rho[i],tt[i]=interf_obl_par(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
# print("Interfaccia:",i," - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
i=nstrati-1
e_p[i] = 1 + 0j;
e_m[i] = 0 + 0j;
#print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
for i in range(nstrati-2, -1, -1):
    e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])

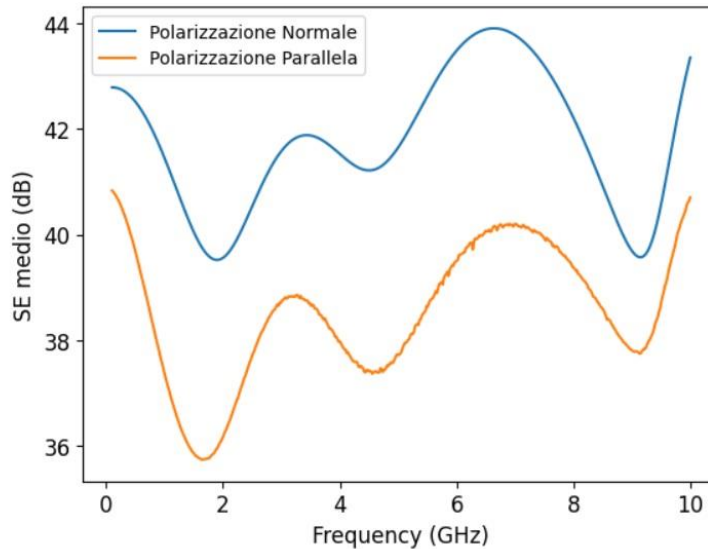
n_angoli = 901
somma_obl_par = 0
for i in range(0,n_angoli - 1,1):
    somma_obl_par = somma_obl_par + 1 / SE_obl_par(freq, i * 0.5 * np.pi / (n_angoli - 1))
somma_obl_par = somma_obl_par / (n_angoli - 1)
y = 20 * np.log10(1 / somma_obl_par) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y, label='Polarizzazione Parallela')
#n_angoli = 181
#somma_obl_par = 0
#for i in range(0,n_angoli - 1,1):
#    somma_obl_par = somma_obl_par + 1 / SE_obl_par(freq, i * 0.5 * np.pi / (n_angoli - 1))
#somma_obl_par = somma_obl_par / (n_angoli - 1)
#y = 20 * np.log10(1 / somma_obl_par) # This is already vectorized, that is, y will be a vector!
#plt.plot(freq*1e-9, y)
#n_angoli = 91

#somma_obl_par = 0
#for i in range(0,n_angoli - 1,1):
#    somma_obl_par = somma_obl_par + 1 / SE_obl_par(freq, i * 0.5 * np.pi / (n_angoli - 1))
#somma_obl_par = somma_obl_par / (n_angoli - 1)
#y = 20 * np.log10(1 / somma_obl_par) # This is already vectorized, that is, y will be a vector!
#plt.plot(freq*1e-9, y)
#n_angoli = 46
#somma_obl_par = 0
#for i in range(0,n_angoli - 1,1):
#    somma_obl_par = somma_obl_par + 1 / SE_obl_par(freq, i * 0.5 * np.pi / (n_angoli - 1))
#somma_obl_par = somma_obl_par / (n_angoli - 1)
#y = 20 * np.log10(1 / somma_obl_par) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y, label='Polarizzazione Parallela')
plt.legend(loc="upper left")
plt.show()

```

I valori individuati sono stati, anche in questo caso, riportati in un grafico dove sull'asse delle ascisse c'è sempre la frequenza in GHz e sulle ordinate l'SE medio in dB.

SE medio per polarizzazione perpendicolare e parallela:



CAPITOLO 3 – Ottimizzazione

Come visto in precedenza, nello sviluppo del codice Python analizzato in questa tesi sono state utilizzate diverse librerie. Tra queste vi è la libreria Scipy: essa in particolare è un pacchetto software per il calcolo scientifico, costruito sopra la libreria Numpy.

Una delle caratteristiche principali di Scipy però, è la sua libreria di ottimizzazione che include funzioni per ridurre al minimo o massimizzare una determinata funzione.

3.1 Funzione SE

Innanzitutto, deve essere implementata una funzione *SE* che verrà usata nel calcolo dell'efficacia di schermatura durante l'ottimizzazione:

```

# Calcolo dell'efficacia di schermatura durante L'ottimizzazione (scegliere la polarizzazione desiderata)
def se(x):
    nstrati=6
    theta_iniziale=85/180*np.pi
    n=[0]*nstrati
    gamma=[0]*nstrati
    eta=[0]*nstrati
    theta_m=[0]*nstrati
    theta_m[0]=theta_iniziale
    tt=[0]*(nstrati-1)
    rho=[0]*(nstrati-1)
    e_p=[0]*(nstrati)
    e_m=[0]*(nstrati)
    #strati = [("aria 1",1,1,0,0),
    #          ("vetro 1",7,1,0.001,0.008264),
    #          ("metallo",1,1,61730000,6.5e-9),
    #          ("aria 2",1,1,0,0.01218),
    #          ("vetro 2",7,1,0.001,0.003615),
    #          ("aria 3",1,1,0,0)]
    strati = [("aria 1",1,1,0,0),
              ("vetro 1",7,1,0.001,x[0]),
              ("metallo",1,1,61730000,x[1]),
              ("aria 2",1,1,0,x[2]),
              ("vetro 2",7,1,0.001,x[3]),
              ("aria 3",1,1,0,0)]
    for i in range(nstrati):
        n[i]=np.sqrt(strati[i][1]*strati[i][2])
        gamma[i],eta[i]=materiale(strati[i])
        # print("Materiale:",strati[i][0], " - Indice di rifrazione =",np.round(n[i]))
        # print("Materiale:", strati[i][0], " - Costante di propagazione =",np.round(gamma[i],2),"(1/m), Impedenza caratteristica =",
    for i in range(1,nstrati,1):
        theta_rifl,theta_m[i]=snell(n[i-1],n[i],theta_m[i-1])
#
#
for i in range(nstrati-1):
    rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
# print("Interfaccia:",i, " - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
i=nstrati-1
e_p[i] = 1 + 0j;
e_m[i] = 0 + 0j;
#print("Materiale:",strati[i][0], " - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
for i in range(nstrati-2, -1, -1):
    e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])
# print("Materiale:",strati[i][0], " - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
#print("SE (dB) = ", np.round(20 * np.Log10(np.abs(e_p[0])),2))
return np.abs(e_p[0])

```

3.2 Calcolo dell'SE dopo l'ottimizzazione

Tra i vari moduli e pacchetti disponibili nella libreria Scipy, è presente il pacchetto *scipy.optimize* che include appunto algoritmi di ottimizzazione. Quando è necessario ottimizzare i parametri in input di una funzione, *scipy.optimize* contiene una serie di metodi utili a tale scopo tra cui *minimize()* che permette di minimizzare una funzione con molte variabili. Questa funzione può gestire diversi input e output e impone dei vincoli sulla soluzione del problema, in particolare tra i vari tipi di vincoli è stato utilizzato il tipo *bounds* che impone un limite inferiore e superiore alla soluzione del problema. Quando vengono utilizzati questi vincoli, viene anche limitata la scelta del metodo di ottimizzazione da utilizzare, poiché non tutti i metodi disponibili supportano tali vincoli.

```

# Punto iniziale e contorni per l'ottimizzazione
x0 = np.array([6e-3, 6e-9, 15e-3, 4e-3])
bnds = (([1e-3, 1e-2], [1e-9, 1e-8], [1e-3, 1e-1], [1e-3, 1e-2])

```

```

# Importazione delle funzioni per l'ottimizzazione
from scipy.optimize import minimize
from scipy.optimize import Bounds

# Ottimizzazione
# LTE
#freq=2.1e9
#freq=1.8e9
#freq=2.6e9
# 5G
freq=3.7e9
res = minimize(se, x0, method='nelder-mead',
               options={'maxiter': 1000}, bounds=bnds, tol=1e-6)

# Stampa gli spessori degli strati ottimizzati
print(res)
print(res.x)

message: Optimization terminated successfully.
success: True
status: 0
      fun: 550.9176954772767
         x: [ 7.091e-03  1.000e-09  1.403e-02  1.000e-03]
        nit: 595
       nfev: 989

```

L'ottimizzazione si occupa dello studio della teoria e dei metodi per la ricerca dei punti di massimo e minimo di una funzione. Dovendo trovare massimi e minimi, la maggior parte degli algoritmi di ottimizzazione noti si basano sul concetto di derivata e sulle informazioni che possono essere dedotte dal gradiente. Tuttavia, molti problemi di ottimizzazione sono caratterizzati dal fatto che l'espressione analitica della funzione obiettivo non è nota ed è quindi impossibile calcolarne le derivate oppure è particolarmente complessa. Per risolvere questo tipo di problemi sono stati sviluppati diversi algoritmi che non tentano di approssimare il gradiente ma piuttosto utilizzano i valori della funzione in un insieme di punti di campionamento per determinare una nuova iterata con altri mezzi. Tra questi algoritmi vi è il *metodo di Nelder Mead* il cui scopo è appunto quello di minimizzare una funzione non lineare senza nessuna informazione sulla sua derivata ma attraverso la sua valutazione in alcuni punti di prova che costituiscono una particolare forma geometrica detta *simplex*.

La funzione `minimize` richiede quindi come argomenti: la funzione da minimizzare, un array di n elementi reali (x_0), il metodo utilizzato, un numero massimo di iterazioni da fare (*maxiter*: 1000), limiti sulle variabili (*bounds*), tolleranza per l'interruzione (*tol*).

A questo punto una volta terminata l'ottimizzazione, può essere nuovamente calcolato l'SE (uguale al calcolo prima dell'ottimizzazione):

```

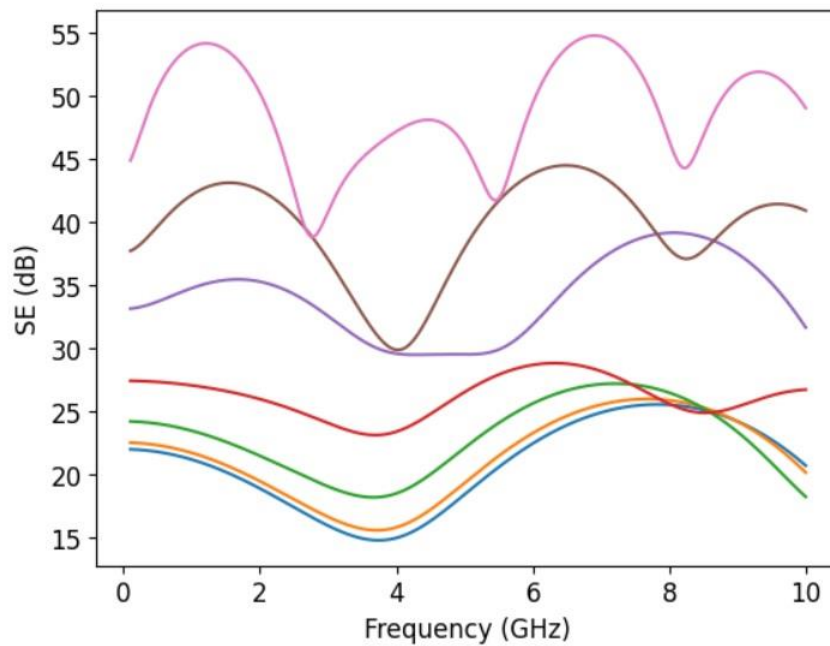
# Calcolo dell'SE dopo l'ottimizzazione
nstrati=6
theta_iniziale=85/180*np.pi
n=[0]*nstrati
gamma=[0]*nstrati
eta=[0]*nstrati
theta_m=[0]*nstrati
theta_m[0]=theta_iniziale
tt=[0]*(nstrati-1)
rho=[0]*(nstrati-1)
e_p=[0]*nstrati
e_m=[0]*nstrati
#strati = [("aria 1",1,1,0,0),
#          ("vetro 1",7,1,0.001,0.008264),
#          ("metallo",1,1,61730000,6.5e-9),
#          ("aria 2",1,1,0,0.01218),
#          ("vetro 2",7,1,0.001,0.003615),
#          ("aria 3",1,1,0,0)]
strati = [("aria 1",1,1,0,0),
          ("vetro 1",7,1,0.001,res.x[0]),
          ("metallo",1,1,61730000,res.x[1]),
          ("aria 2",1,1,0,res.x[2]),
          ("vetro 2",7,1,0.001,res.x[3]),
          ("aria 3",1,1,0,0)]
#freq=1e9
freq = np.linspace ( start = 100e6      # Lower limit (Hz)
                    , stop = 10e9      # upper limit (Hz)
                    , num = 501        # generate 501 points between 0 and 3
                    )
for i in range(nstrati):
    n[i]=np.sqrt(strati[i][1]*strati[i][2])
    gamma[i],eta[i]=materiale(strati[i])
    # print("Materiale:",strati[i][0]," - Indice di rifrazione =",np.round(n[i]))
    # print("Materiale:",strati[i][0]," - Costante di propagazione =",np.round(gamma[i],2),"(1/m), Impedenza caratteristica =",np.round(eta[i]
for i in range(1,nstrati,1):
    theta_rifl,theta_m[i]=snell(n[i-1],n[i],theta_m[i-1])

for i in range(nstrati-1):
    rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
    # print("Interfaccia:",i," - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
i=nstrati-1
e_p[i] = 1 + 0j;
e_m[i] = 0 + 0j;
#print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
for i in range(nstrati-2, -1, -1):
    e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])
    # print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
#print("SE (dB) = ", np.round(20 * np.Log10(np.abs(e_p[0])),2))
#fstart = 0.1 # IN GHz
#fstop = 10 # IN GHz
#fstep = 1e-3 # IN GHz
#fnum = (fstop - fstart) / fstep + 1

plt.rc('xtick', labelsize=12)
plt.rc('ytick', labelsize=12)
plt.rc('axes', labelsize=12)
plt.xlabel('Frequency (GHz)')
plt.ylabel('SE (dB)')
#
# scegliere se calcolare 'SE_obl_perp' o 'SE_obl_par'
#
y = 20 * np.log10(SE_obl_perp(freq,0/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,15/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,30/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,45/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,60/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,67.5/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
y = 20 * np.log10(SE_obl_perp(freq,75/180*np.pi)) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y)
plt.show()

```

In questo caso in grafico sarà:



3.3 SE medio dopo l'ottimizzazione

A questo punto una volta completata l'ottimizzazione e aver calcolato i valori dell'SE al variare della frequenza (e dell'angolo d'incidenza), anche in questo caso sono stati individuati un SE medio per la polarizzazione parallela e per quella perpendicolare:

```
# Grafico dell'SE dopo l'ottimizzazione
freq = np.linspace ( start = 100e6 # Lower Limit (Hz)
                    , stop = 10e9 # upper limit (Hz)
                    , num = 501 # generate 501 points between 0 and 3
                    )
for i in range(nstrati):
    n[i]=np.sqrt(strati[i][1]*strati[i][2])
    gamma[i],eta[i]=materiale(strati[i])
    # print("Materiale:",strati[i][0]," - Indice di rifrazione =",np.round(n[i]))
    # print("Materiale:",strati[i][0]," - Costante di propagazione =",np.round(gamma[i],2),"(1/m), Impedenza caratteristica =",np.round(eta[i])
    for i in range(1,nstrati,1):
        theta_rifl,theta_m[i]=snell(n[i-1],n[i],theta_m[i-1])
        # print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"V/m")
        #print("SE (dB) = ", np.round(20 * np.Log10(np.abs(e_p[0])),2))
        #fstart = 0.1 # IN GHz
        #fstop = 10 # IN GHz
        #fstep = 1e-3 # IN GHz
        #fnum = (fstop - fstart) / fstep + 1

        #print("x = ",x)
        #print("y = ",y)
        plt.rc('xtick', labels=12)
        plt.rc('ytick', labels=12)
        plt.rc('axes', labels=12)
        plt.xlabel('Frequency (GHz)')
        plt.ylabel('SE medio (dB)')
    for i in range(nstrati-1):
        rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
        # print("Interfaccia:",i," - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
    i=nstrati-1
    e_p[i] = 1 + 0j;
    e_m[i] = 0 + 0j;
    #print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"V/m")
    for i in range(nstrati-2, -1, -1):
        e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])
```

```

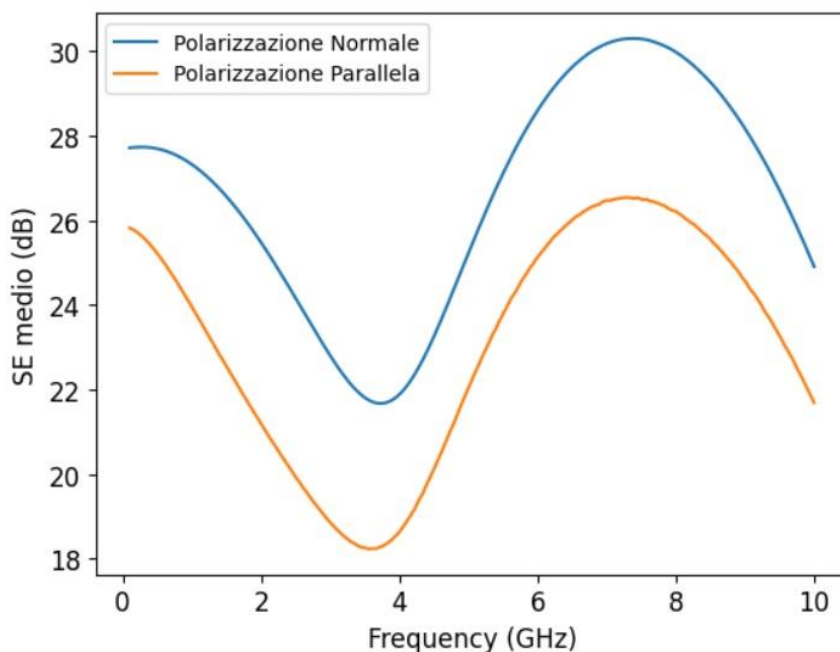
n_angoli = 181
somma_obl_perp = 0
for i in range(0,n_angoli - 1,1):
    somma_obl_perp = somma_obl_perp + 1 / SE_obl_perp(freq, i * 0.5 * np.pi / (n_angoli - 1))
somma_obl_perp = somma_obl_perp / (n_angoli - 1)
y = 20 * np.log10(1 / somma_obl_perp) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y, label='Polarizzazione Normale')
for i in range(nstrati-1):
    rho[i],tt[i]=interf_obl_perp(eta[i],eta[i+1],theta_m[i-1],theta_m[i])
# print("Interfaccia:",i," - Coefficiente di trasmissione =",np.round(tt[i],5)," Coefficiente di riflessione =",np.round(rho[i],5))
i=nstrati-1
e_p[i] = 1 + 0j;
e_m[i] = 0 + 0j;
#print("Materiale:",strati[i][0]," - E progressivo =",np.round(e_p[i],2),"(V/m), E regressivo =",np.round(e_m[i],2),"(V/m)")
for i in range(nstrati-2, -1, -1):
    e_p[i], e_m[i] = campi (gamma[i+1], strati[i+1][4]/np.cos(theta_m[i+1]), rho[i], tt[i], e_p[i+1], e_m[i+1])

n_angoli = 901
somma_obl_par = 0
for i in range(0,n_angoli - 1,1):
    somma_obl_par = somma_obl_par + 1 / SE_obl_par(freq, i * 0.5 * np.pi / (n_angoli - 1))
somma_obl_par = somma_obl_par / (n_angoli - 1)
y = 20 * np.log10(1 / somma_obl_par) # This is already vectorized, that is, y will be a vector!
plt.plot(freq*1e-9, y, label='Polarizzazione Parallela')

plt.legend(loc="upper left")
plt.show()

```

Il grafico risultate con le frequenze sull'asse delle ascisse in GHz e l'SE medio in dB sulle ordinate sarà:



Conclusioni

Partendo da un programma iniziale per il calcolo dell'efficacia di schermatura, è stato possibile trovare un codice ottimizzato, tramite opportune funzioni e librerie fornite dal linguaggio Python, che funziona correttamente.

Si è visto che il rivestimento metallico inserito all'interno delle vetrate comporta un notevole peggioramento delle capacità trasmissive del mezzo, riscontrando una differenza che si aggira intorno ai 30 dB tra il caso in cui tale rivestimento è presente e quello in cui è assente.

In termini di efficacia di schermatura è importante la scelta del materiale utilizzato per la realizzazione del rivestimento in quanto la capacità schermante è dovuta principalmente ai valori di conducibilità elettrica del metallo scelto. Sulla base di queste osservazioni è lecito affermare che la scelta migliore è quella di utilizzare un metallo con conducibilità elettrica relativamente bassa.

In questa tesi è stato analizzato il caso in cui l'onda elettromagnetica incide in modo obliquo sulla superficie (l'unità a doppio vetro) quindi che presenta un certo angolo d'incidenza, e si è potuto constatare attraverso la consultazione dei grafici ottenuti che maggiore è l'angolo con cui l'onda incide sul mezzo, maggiore sarà l'attenuazione del campo. In particolare, all'aumentare dell'angolo d'incidenza aumenta l'efficacia di schermatura, tanto più quando ci si avvicina alla situazione limite ovvero quando l'angolo d'incidenza è di 90° rispetto alla normale al piano del vetro. In quest'ultimo caso l'onda non può attraversare la vetrata e di conseguenza non vi è in alcun modo una trasmissione di campo all'interno dell'edificio.

Bibliografia

http://microwave.unipv.it/pages/campi_pg/docs/Campi_EM_Lezione_04_onde_piane.pdf

https://www.beshielding.it/wp-content/uploads/2020/10/beshielding_schermatura_elettromagnetica_elf_it_en-min.pdf

http://docenti.ing.unipi.it/p.nepa/Radiazioni_elettromagnetiche_biomedica/Materiale_didattico_Genovesi/Note_sulla_Shielding_Effectiveness.pdf

<https://www.html.it/pag/15608/perche-usare-python/>

<https://www.programmareinpython.it/video-corso-python-base/scrivi-le-tue-funzioni/>

<https://realpython.com/python-scipy-cluster-optimize/>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html>

Moglie Franco, Primiani Mariani Valter, Bastianelli Luca, 2021, articolo *Shielding effectiveness Controlling of Coated Glass Assembly in Mobile and Positioning GPS Frequency Bands*

Fawwaz T. Ulaby, *Fondamenti di campi elettromagnetici*

Tesi di Laura Marinaro, *Efficacia di schermatura di vetri compositi per la protezione di ambienti e persone da campi elettromagnetici.*

Tesi di Tomasz De Clerico, *Ottimizzazione della geometria di materiali per la schermatura elettromagnetica di ambienti civili.*

Tesi di Julian Gostoli, *Sviluppo di un codice in Python per l'analisi dell'efficacia di schermatura di materiali stratificati*

Ringraziamenti

Al termine di questo elaborato, vorrei ringraziare tutte le persone che mi hanno sostenuto durante tutto il mio percorso universitario e senza le quali tutto questo sarebbe stato sicuramente più difficile.

Per prima cosa vorrei ringraziare il mio relatore Prof. Franco Moglie e correlatore Prof. Valter Mariani Primiani, che mi hanno seguito con molta disponibilità nella stesura di questa tesi e in tutto il percorso di tirocinio.

Un ringraziamento speciale va alla *mia famiglia*, senza il loro continuo sostegno non sarei mai riuscita ad arrivare a questo importante traguardo. Grazie per avermi sempre spronata, aver appoggiato ogni mia decisione e avermi aiutata nei tanti momenti di sconforto durante questo percorso. Voglio ringraziarvi per il vostro amore e incoraggiamento, e per avermi sempre fatto sentire quanto voi crediate in me. Questa tesi la dedico a voi, con la speranza di avervi reso fieri di me.

A *Elio*, sempre al mio fianco in ogni mia scelta. Grazie per aver sopportato le mie ansie e paure e aver sempre cercato di tranquillizzarmi. Anche se lontani, abbiamo camminato insieme in questo percorso e senza mai lasciarmi sola mi hai fatto capire ogni giorno quanto tu credessi in me.

A *Giacomo*, in questi anni ci siamo sempre sostenuti l'un l'altra, ci siamo incoraggiati affrontando insieme questo cammino passo dopo passo. Non ti ringrazierò mai abbastanza per l'aiuto e il sostegno, per essere stato al mio fianco in ogni momento, per come riesci a capirmi senza che io dica una parola, ma soprattutto ti ringrazierò per sempre per l'amico che sei. Non ce lo diciamo mai a parole, ma sappiamo quanto è forte il bene che ci lega e che spero duri per tutta la vita.

Alla mia coinquilina *Miriam* con la quale in questi tre anni universitari ho condiviso momenti di gioia, successi ma anche insuccessi. Ci conosciamo da tanti anni, ma anche se un po' in ritardo sono felice di aver trovato in te un'amica sincera e sempre pronta ad aiutarmi. Grazie per avermi fatto da sveglia ogni giorno, senza di te sicuramente oggi avrei avuto ancora molti esami

da dare.

A Francesca, Alice e Marialaura. La nostra amicizia è nata sui banchi di scuola, insieme a voi ho condiviso alcuni dei momenti migliori della mia vita. Abbiamo riso, gioito, viaggiato, litigato ma siamo sempre riuscite a ritrovarci. Anche se le nostre strade universitarie si sono divise, anche a centinaia di chilometri di distanza so che potrò sempre contare su di voi.

A Giorgia, Chiara e Francesca, le mie “commare”. Ci siamo conosciute in una sala di danza e man mano con il tempo abbiamo stretto un legame speciale che ci tiene legate ancora oggi a distanza di anni. Grazie per ogni momento passato insieme e per avermi sostenuta e aiutata come delle vere sorelle maggiori.

A Erika, Francesca, Elisabeth e Veronica, le mie amiche di sempre. Il nostro è un rapporto speciale, siamo cresciute insieme come delle vere sorelle condividendo in questi venti anni ogni momento delle nostre vite. Nonostante le nostre strade “scolastiche” si siano divise, la nostra amicizia e sorellanza non è mai cambiata e spero che non cambierà mai.

Ringrazio infine tutti voi amici che mi siete stati accanto in questi anni, chi aiutandomi con la preparazione degli esami e chi semplicemente con la loro presenza, e che siete presenti anche oggi per questo mio importante traguardo. Vi voglio bene a tutti.