



UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

Dipartimento di Fisica
CORSO DI LAUREA IN FISICA
TESI DI LAUREA

Analisi di sistemi ottici composti,
mediante il metodo matriciale dei raggi
parassiali e ray tracing

Laureando:
Riccardo Corte

Relatore:
Prof. Edoardo Milotti

Correlatore:
Dott. Giacomo Principe

*Ai sorrisi di chi mi vuole bene,
alle risate della Zia della Mamma
e alla Nonna.*

Sommario

L'obiettivo di questa tesi è applicare due strumenti di studio avanzati per caratterizzare il fenomeno delle aberrazioni ottiche in sistemi di lenti composte. Il metodo matriciale per raggi parassiali è il primo strumento presentato. Questo metodo è caratterizzato dalla possibilità di descrivere i sistemi ottici a partire da moduli matriciali elementari. Ciò determina un'elevata versatilità che verrà sfruttata per descrivere sia sistemi semplici, come lenti spesse, che sistemi complessi come il tripletto di Cooke. Il limite della rappresentazione matriciale dei sistemi ottici risiede nelle assunzioni alla base del metodo, valido solo per i raggi parassiali. A completamento del metodo matriciale viene utilizzato il ray tracing, un metodo più complesso da applicare ma che fornisce risultati più accurati. Per applicare il ray tracing è stato elaborato un programma in Python che ha permesso di rappresentare in maniera completa i sistemi già analizzati matematicamente applicando il metodo matriciale. Il confronto fra i due metodi ha evidenziato il fenomeno delle aberrazioni ottiche, che consiste nello scostamento tra l'immagine proiettata da un piccolo foro, non tenendo conto del fenomeno della diffrazione, e quella prodotta dal sistema ottico completo. Sono state così caratterizzate le aberrazioni sferiche e le aberrazioni cromatiche dei sistemi.

Indice

1	Introduzione generale	1
2	Metodo matriciale per raggi parassiali	3
2.1	Aspetti generali del metodo	4
2.2	Applicazione del metodo alle lenti spesse	7
2.2.1	Proprietà delle lenti spesse	8
2.2.2	Esempio per una lente spessa	13
2.3	Applicazione del metodo ai sistemi composti	14
2.3.1	Proprietà del tripletto di Cooke	16
2.4	Aberrazioni cromatiche e metodo matriciale per raggi parassiali	18
3	Il metodo del ray tracing	20
3.1	Descrizione generale del metodo	20
3.2	Applicazione del metodo per una lente spessa	23
3.3	Applicazione del metodo al tripletto di Cooke	26
3.4	Aberrazioni cromatiche e ray tracing	31
4	Conclusioni	38
5	Bibliografia	40
A	Appendice	41
A.1	Codice Python per il ray tracing del tripletto di Cooke	41
A.2	Rappresentazione matriciale del tripletto di cooke	60

1 Introduzione generale

Lo studio effettuato si pone l’obiettivo di applicare i principi dell’ottica geometrica per rappresentare e analizzare sistemi ottici composti.

Lo studio di sistemi ottici complessi può essere effettuato utilizzando il formalismo matriciale per raggi parassiali (Cap.3). Il metodo matriciale per raggi parassiali è uno strumento molto potente e versatile in quanto consente di definire dei moduli che una volta applicati ai sistemi di lenti permettono di descrivere qualsiasi sistema ottico nella forma di una matrice 2×2 (due per due). I limiti di questo metodo sono legati alle sue approssimazioni, fondamentali alla rappresentazione degli elementi ottici come matrici due per due. Queste approssimazioni semplificano il sistema reale, mascherando le imperfezioni fisiche caratteristiche delle lenti. I risultati ottenuti per sistemi reali si discostano dalla rappresentazione ideale propria dall’ottica matriciale generando il fenomeno delle aberrazioni. Fra le approssimazioni alla base del metodo vi è l’approssimazione per raggi parassiali, o per piccoli angoli. Consiste nell’ipotizzare che gli angoli di incidenza con la lente siano sufficientemente piccoli da rendere valida:

$$\sin \theta \approx \tan \theta \approx \theta \quad (1)$$

Come riportato in [2], l’approssimazione parassiale assume che l’angolo sia piccolo. Tale approssimazione determina che solo il primo termine dell’espansione di Taylor delle funzioni trigonometriche venga considerato. L’errore introdotto risulta trascurabile solo per angoli inferiori a ($\simeq 20^\circ$). Questa approssimazione elimina la necessità di utilizzare le funzioni trigonometriche, così come la necessità di espandere le equazioni differenziali al secondo ordine nella descrizione della propagazione delle onde luminose.

In questa tesi si intende utilizzare il metodo matriciale sia per la descrizione di oggetti semplici come le lenti spesse, sia per oggetti più complessi come gli obiettivi fotografici. L’efficacia dell’approccio matriciale verrà testata applicando il metodo ad un obiettivo fotografico classico, il Tripletto di Cooke.

Il tripletto di Cooke, ha rappresentato il primo e più semplice schema di assemblaggio di tre lenti in grado di correggere tutte le imperfezioni principali degli obiettivi fotografici, costituite dalle aberrazioni di primo e terzo ordine. Tutt’oggi trova numerose applicazioni nello sviluppo di sensori per la rilevazione di dati biometrici. Come è possibile osservare in Fig.(1), si costituisce di tre elementi, tre lenti spesse, due convesse ed una concava. Le lenti esterne sono due lenti positive in vetro crown, mentre l’elemento centrale è una lente negativa in vetro flint¹.

Il motivo per cui si sono creati questi sistemi composti, è il maggior numero di gradi di libertà che posseggono. Queste variabili consentono di modificare il sistema e perfezionare la convergenza dei raggi eliminando il fenomeno delle aberrazioni. In particolare il tripletto di Cooke è costituito da otto variabili indipendenti efficaci: le sei superfici delle lenti ed i due spazi di aria fra gli elementi.

¹Per maggiori informazioni sulle tipologie di vetro interne agli obiettivi fotografici ci si rifà a [11]

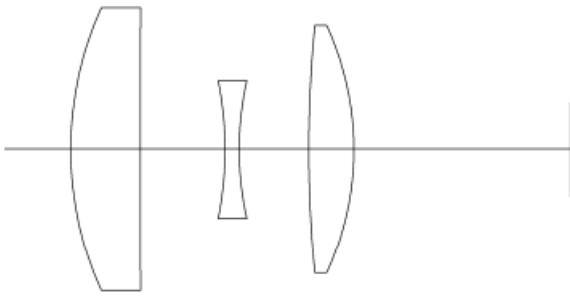


Figura 1: Rappresentazione schematica del tripletto di Cooke.

A completamento del metodo matriciale si applicherà il ray tracing (Cap.3), un metodo capace di rappresentare esattamente i sistemi descritti dall’ottica geometrica. Il ray tracing, ”tracciamento dei raggi”, simula il percorso effettivo del raggio attraverso il sistema ottico per qualsiasi raggio in entrata. A partire dall’applicazione della Legge di Snell è possibile ottenere valori non approssimati degli angoli di rifrazione all’interfaccia fra due superfici, e quindi determinare il percorso reale del raggio. Si mostrerà anche come il metodo del ray tracing sia efficace nell’analisi delle aberrazioni sferiche, caratteristiche del sistema complesso appena descritto.

Si è già accennato come le aberrazioni rappresentino deviazioni dalla perfezione ideale nella formazione delle immagini. Queste imperfezioni possono compromettere la qualità dell’immagine. La correzione delle aberrazioni risulta quindi fondamentale per la realizzazione di sistemi fotografici di qualità.

Lo studio verterà sull’analisi delle aberrazioni sferiche monocromatiche e le aberrazioni sferiche cromatiche. Le aberrazioni sferiche, Fig.(2) si originano quando i raggi luminosi che attraversano una lente sferica non convergono in un unico punto focale, dando luogo a immagini sfocate. Questo fenomeno si verifica principalmente a causa della diversa angolazione con cui i raggi luminosi incidono sulle superfici sferiche delle lenti, che porta ad una variazione nella posizione di intersezione con l’asse ottico.

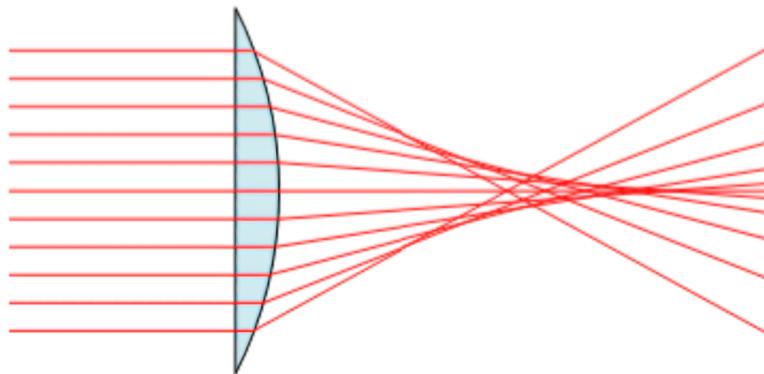


Figura 2: Fenomeno delle aberrazioni ottiche per una lente spessa.

L'aberrazione cromatica è un fenomeno ottico che si manifesta quando una lente non è in grado di focalizzare tutte le lunghezze d'onda della luce nello stesso punto. Questo accade perché l'indice di rifrazione del materiale varia in funzione della lunghezza d'onda. Di conseguenza, la luce bianca, composta da diverse lunghezze d'onda, si separa nelle sue componenti spettrali, creando bordi colorati attorno alle immagini. L'aberrazione cromatica sferica si verifica quando i vari colori non vengono messi a fuoco nello stesso piano lungo l'asse ottico, Fig.(3). Per esempio i raggi di luce blu, solitamente i più rifratti, convergono prima dei raggi di luce rossa, causando sfocature e frange colorate nelle immagini. Per ridurre questi effetti indesiderati, vengono utilizzati sistemi di lenti acromatiche o apocromatiche, composte da più materiali con indici di rifrazione diversi, capaci di correggere parzialmente o completamente queste aberrazioni, un esempio di queste tipologie di lenti è il tripletto di Cooke.

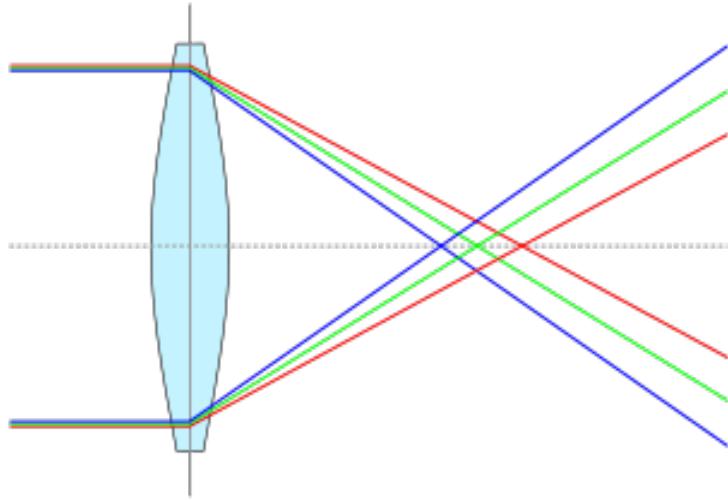


Figura 3: Fenomeno delle aberrazioni cromatiche per una lente spessa.

2 Metodo matriciale per raggi parassiali

Per caratterizzare i sistemi composti di nostro interesse, come quello mostrato in Fig.(1) verrà utilizzata l'ottica matriciale. L'ottica matriciale è la tecnica specifica per il tracciamento dei raggi parassiali Eq.(1). Un raggio si può descrivere a partire dalla sua posizione ed il suo angolo rispetto all'asse ottico. Nell'approssimazione dei raggi parassiali, la posizione e l'angolo del raggio incidente al sistema ottico sono legati linearmente all'angolo e alla posizione del raggio rifratto. La proporzione lineare fra i valori iniziali e finali permette di definire delle matrici 2×2 (due per due) utili alla descrizione del sistema considerato. Queste matrici vengono anche chiamate matrici $ABCD$ in ragione dei quattro elementi di cui sono composte [3].

Nell'ipotesi dei raggi parassiali, la relazione fra le coordinate del raggio incidente (y_1, θ_1) ed il raggio rifratto (y_2, θ_2) vale:

$$y_2 = Ay_1 + B\theta_1$$

$$\theta_2 = Cy_1 + D\theta_1$$

e si può quindi riscrivere nella forma:

$$\begin{bmatrix} y_2 \\ \theta_2 \end{bmatrix} = \underbrace{\begin{bmatrix} A & B \\ C & D \end{bmatrix}}_{M} \cdot \begin{bmatrix} y_1 \\ \theta_1 \end{bmatrix} \quad (2)$$

Con A,B,C,D parametri reali. Si è quindi compreso il processo attraverso cui verranno definite le matrici $ABCD$ usate per descrivere gli elementi ottici trattati.

2.1 Aspetti generali del metodo

Il metodo matriciale si basa sulla descrizione di matrici di componenti ottici elementari che possono essere combinate per descrivere sistemi più complessi. Procediamo dunque ad identificare quali sono gli elementi ottici che andranno a costituire le matrici fondamentali di questo metodo.

- Propagazione in un mezzo omogeneo.

All'interno di un mezzo avente indice di rifrazione omogeneo n_i , i raggi si propagano percorrendo il cammino ottico estremale², vedi Fig.(4).

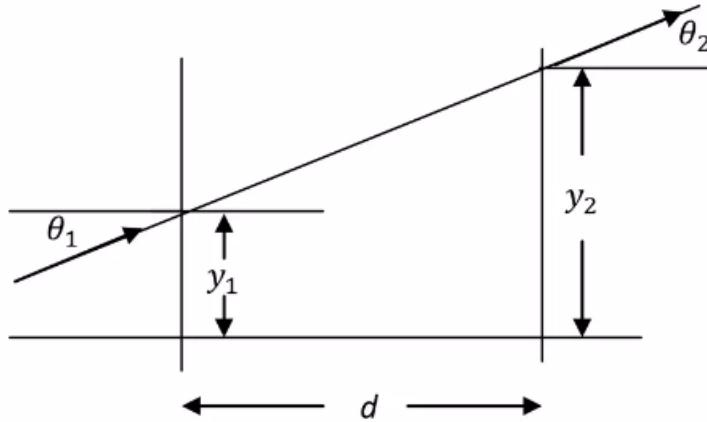


Figura 4: Propagazione in un mezzo.

Un raggio che percorre una distanza d in un mezzo avente indice di propagazione n determina la matrice:

$$M = \begin{bmatrix} 1 & d \\ 0 & 1 \end{bmatrix} \quad (3)$$

- Rifrazione su una superficie piana.

²Anche conosciuto come principio di Fermat [3]

In corrispondenza di una superficie piana fra due mezzi aventi indice di rifrazione n_1 , n_2 l'angolo rifratto varia in accordo alla legge di Snell, e la matrice del raggio trasferito è la seguente.

$$M = \begin{bmatrix} 1 & 0 \\ 0 & \frac{n_1}{n_2} \end{bmatrix} \quad (4)$$

- Rifrazione su una superficie sferica.

Si descrive ora il fenomeno della rifrazione all'interfaccia sferica fra due mezzi, vedi Fig.(5); per una trattazione completa del procedimento si consulti [3]. Applicando la legge di Snell, si ottengono le relazioni tra y_1 , y_2 e θ_1 , θ_2 , in funzione degli indici di rifrazione iniziale n_1 , finale n_2 , e del raggio della superficie R .

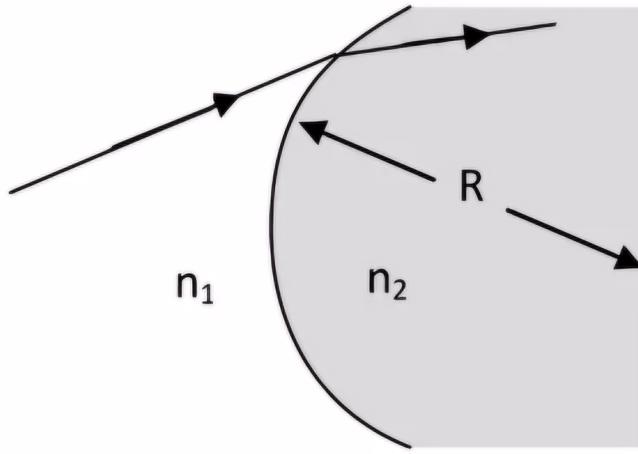


Figura 5: Rifrazione su una superficie sferica

$$\begin{aligned} y_2 &= \underbrace{\frac{1}{A}}_{y_1} + \underbrace{\frac{0}{B}}_{\theta_1} \theta_1 \\ \theta_2 &= \underbrace{\frac{(n_1 - n_2)}{n_2 R}}_C y + \underbrace{\frac{n_1}{n_2}}_D \theta_1 \end{aligned} \quad (5)$$

Si esprime così la matrice che lega l'angolo rifratto θ_2 per un angolo incidente θ_1 , nell'approssimazione dei raggi parassiali

$$M = \begin{bmatrix} 1 & 0 \\ \frac{(n_1 - n_2)}{n_2 R} & \frac{n_1}{n_2} \end{bmatrix} \quad (6)$$

- Trasmissione attraverso una lente sottile.

La relazione tra θ_1 e θ_2 per raggi parassiali trasmessi attraverso una lente sottile viene espressa in funzione della distanza focale f della lente, viene qui riportata,

$$M = \begin{bmatrix} 1 & 0 \\ -\frac{1}{f} & 1 \end{bmatrix} \quad (7)$$

- Riflessione su uno specchio piano

A seguito della riflessione su di uno specchio piano la posizione e l'angolo di riflessione rimangono invariati, la matrice di trasferimento del raggio coincide dunque con la matrice identità.

$$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (8)$$

I sistemi matriciali elementari appena descritti saranno quindi alla base della creazione delle matrici dei sistemi composti.

Nei sistemi più complessi, composti da più elementi, la matrice totale è data dalla moltiplicazione tra le matrici delle varie componenti. Una serie di più elementi ottici M_i viene anche chiamata "cascata" [3]. La matrice M risultante di un sistema costituito da una cascata di N elementi ottici, in ordine: M_1, M_2, \dots, M_N , è ottenuta moltiplicando le singole matrici tra loro, seguendo la regola: $M = M_N \cdots M_2 M_1$.

Si vede ora un'interessante proprietà dei sistemi matriciali che potrà essere usata per verificare i risultati che si otterranno. Un'analisi sul valore del determinante della matrice di un sistema composto fornisce un risultato valido per qualsiasi sistema. Si nota la seguente differenza: il determinante della matrice in Eq.(6), la quale descrive la transizione attraverso un interfaccia fra un mezzo con indice rifrazione n_1 ad un mezzo con indice di rifrazione n_2 , ritorna come risultato $\frac{n_1}{n_2}$, mentre il determinante della matrice in Eq. (3), dove si descrive un sistema con indice di rifrazione costante, vale uno.

Dall'unione dei due sistemi si ottiene una matrice che è data dalla moltiplicazione delle matrici in Eq. (3), Eq. (6), di conseguenza è possibile calcolare il determinante della matrice totale che è dato dal prodotto dei determinanti delle singole matrici. Si ricava che il determinante di una matrice, rappresentante un sistema composto da due elementi, è dato dal rapporto fra gli indici di rifrazione dei mezzi in cui gli elementi sono immersi [3]. Otteniamo dunque un risultato generale per una matrice totale M composta da due elementi.

$$(y_f, \theta_f) = M \cdot (y_i, \theta_i) ; \quad M = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \quad (9)$$

$$\|M\| = \frac{n_1}{n_2} \quad (10)$$

Questo risultato è espandibile per iterazione ad un numero indefinito di elementi. Per un sistema di più elementi si ottiene $\|M\| = \frac{n_i}{n_f}$, con n_i e n_f indici di rifrazione iniziale e finale.

Si è dunque ottenuto un interessante risultato per il valore del determinante della matrice ($\|M\|$) per una lente immersa in aria. In tale caso l'indice di rifrazione del mezzo iniziale e finale si equivalgono, dunque si ottiene $\|M\| = 1$. Nella pratica questo risultato permette di controllare la correttezza delle matrici, e risulta particolarmente utile nel caso di rappresentazioni matriciali particolarmente complesse.

2.2 Applicazione del metodo alle lenti spesse

La descrizione matriciale delle lenti spesse inizia dalla identificazione delle componenti che descrivono il sistema completo. Come mostrato in Fig.(6), una lente spessa si compone di una superficie sferica iniziale (1), un corpo centrale(2) ed una superficie sferica finale(3).

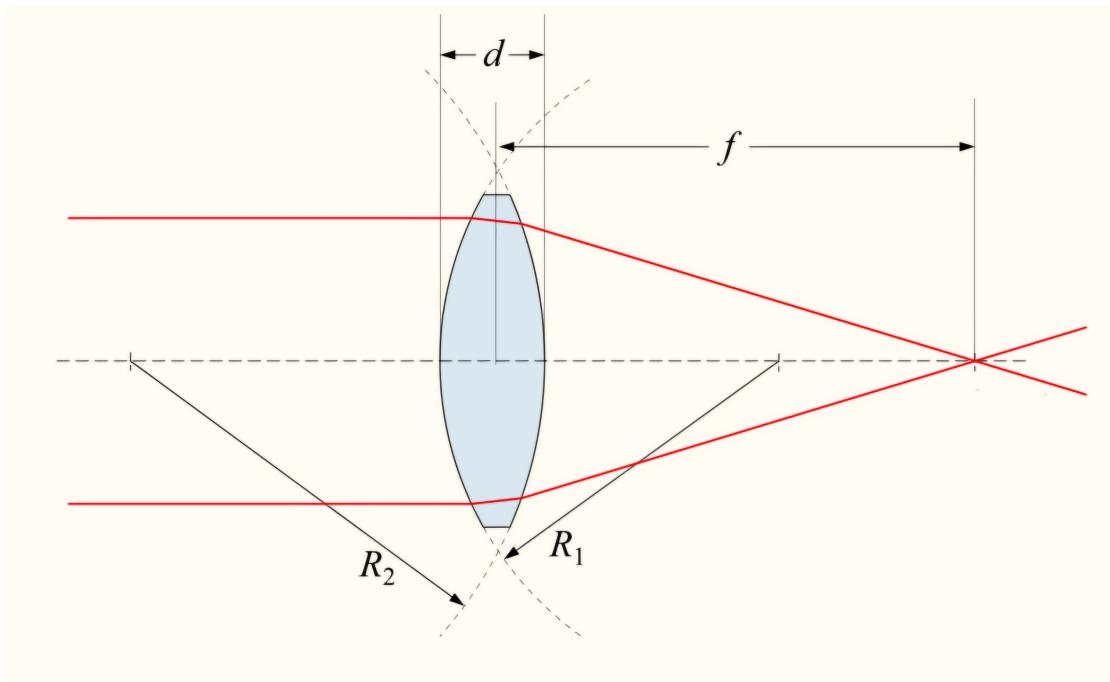


Figura 6: Schema di una lente spessa

- Le matrici che descrivono le superfici sferiche sono già state caratterizzate nel capitolo precedente. Applichiamo dunque Eq.(6) per la prima superficie di una lente spessa avente n_1 indice di rifrazione del mezzo in cui è immersa, n_2 indice di rifrazione del vetro della lente e R_1 raggio di curvatura della prima superficie della lente.

$$M_1 = \begin{bmatrix} 1 & 0 \\ \frac{(n_1-n_2)}{n_2 R_1} & \frac{n_1}{n_2} \end{bmatrix} \quad (11)$$

2. La seconda componente di una lente spessa consiste nello spessore del vetro, applicando Eq.(3) per un vetro avente indice di rifrazione n_2 e spessore d , si ottiene:

$$M_2 = \begin{bmatrix} 1 & \frac{d}{n_2} \\ 0 & 1 \end{bmatrix}$$

3. La seconda superficie sferica è concava. Questo non determina cambiamenti per quanto concerne la rappresentazione matriciale, ed influirà nel calcolo solo al momento della sostituzione dei valori specifici della lente, dove il segno del raggio R_2 sarà negativo. Si applica dunque la matrice Eq.(6), per un indice di rifrazione iniziale n_2 , finale n_3 e raggio di curvatura R_2 ottenendo:

$$M_3 = \begin{bmatrix} 1 & 0 \\ \frac{(n_2-n_3)}{n_3 R_2} & \frac{n_2}{n_3} \end{bmatrix}$$

Si conclude dunque effettuando le moltiplicazioni fra matrici secondo quanto indicato in Eq. 6. Si ottiene così l'espressione matriciale per una lente spessa:

$$M_{ls} = \begin{bmatrix} 1 & 0 \\ \frac{(n_2-n_3)}{n_3 R_2} & \frac{n_2}{n_3} \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{d}{n_2} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ \frac{(n_1-n_2)}{n_2 R_1} & \frac{n_1}{n_2} \end{bmatrix}$$

$$M_{ls} = \begin{bmatrix} 1 + \frac{d(n_1-n_2)}{n_2^2 R_1} & \frac{dn_1}{n_2^2} \\ \frac{(n_1-n_2)\left(\frac{n_2}{n_3} + \frac{d(n_2-n_3)}{n_2 n_3 R_2}\right)}{n_2 R_1} + \frac{(n_2-n_3)}{n_3 R_2} & \frac{n_1\left(\frac{n_2}{n_3} + \frac{d(n_2-n_3)}{n_2 n_3 R_2}\right)}{n_2} \end{bmatrix} \quad (12)$$

Per verificare la correttezza del risultato ottenuto procediamo a calcolare il determinante della matrice in Eq.(12).

$$\|M_{ls}\| = \frac{n_1}{n_3} \quad (13)$$

Che vediamo essere coerente col risultato di Eq.(10), infatti per una lente spessa immersa in aria si ha:

$$\|M_{ls}\| = 1 \quad \square$$

2.2.1 Proprietà delle lenti spesse

La lente spessa così ottenuta rappresenta un risultato grezzo, per ottenere i valori utili alla rappresentazione dei sistemi ottici è necessario espandere ulteriormente le analisi.

La proprietà principale dei sistemi ottici è la capacità di focalizzare i raggi entranti in un punto ben definito. Questo punto è chiamato punto focale e la sua distanza dal vertice della lente definisce la distanza focale. Con riferimento a Fig.(7) è stata ricavata l'espressione matematica della distanza

focale in funzione dei parametri della lente. La distanza focale, coincide con la distanza fra il vertice della lente ed il punto in cui un raggio, entrante parallelamente all'asse ottico converge a seguito della rifrazione della lente. La rappresentazione gaussiana della distanza focale f è la seguente:

$$\frac{1}{s_0} + \frac{1}{s_1} = \frac{1}{f} \quad (14)$$

Dove s_0 è la distanza di un generico oggetto dal centro ottico mentre s_1 è la distanza della sua immagine Fig.(7).

É possibile inoltre esprimere la distanza focale in funzione del raggio di curvatura iniziale R_1 , finale R_2 e dello spessore d della lente.

$$\frac{1}{f} = (n_1 - n_0) \left[\frac{1}{R_1} - \frac{1}{R_2} + \frac{(n_1 - n_0)d}{n_1 R_1 R_2} \right] \quad (15)$$

In questo caso n_0 è l'indice di rifrazione del mezzo in cui è immersa la lente mentre n_1 è l'indice di rifrazione del vetro.

Nel caso di sistemi ottici complessi, come le lenti spesse, i punti focali non sono sufficienti a descrivere completamente le caratteristiche del sistema. É infatti possibile effettuare delle ulteriori approssimazioni sul percorso dei raggi, queste semplificazioni, conseguenza delle proprietà delle lenti, definiscono altri punti utili alla definizione della zona in cui i raggi vengono focalizzati. L'insieme dei punti, che descrivono i sistemi ottici vengono detti punti cardinali. Oltre ai punti focali fanno parte di questo insieme di punti anche i "punti principali". I punti principali P_1, P_2 Fig.(7) sono due punti giacenti sull'asse ottico che permettono di trattare la lente come se tutte le rifrazioni avvenissero sui piani principali³. Come mostrato in Fig.(7) la lente spessa si comporta come se il raggio entrante, che intercetta il primo piano principale, venisse trasferito al secondo mantenendo invariata la sua altezza rispetto all'asse ottico.

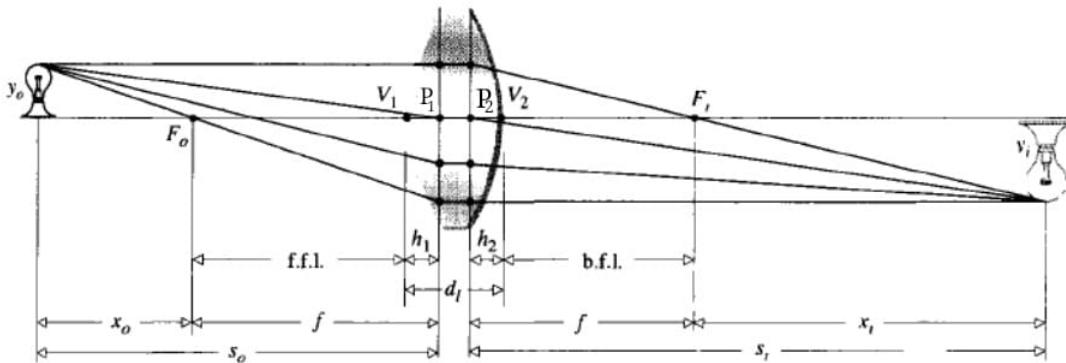


Figura 7: Schema rappresentativo dei punti cardinali

³I due piani principali sono i piani normali all'asse ottico, passanti per i punti principali

È possibile semplificare il comportamento dei sistemi di lenti complesse assimilandoli a lenti sottili semplici. Tale semplificazione avviene calcolando la propagazione dei raggi fino ai punti principali che andranno a costituire i vertici della zona di lente che verrà trattata come una lente semplificata. Rifacendoci a Fig.(7) si può dimostrare che per i punti principali $P_{1,2}$, in una lente i cui vertici sono indicati con V_i , avente distanza focale f , la distanza $(\overline{P_i V_i})$ vale:

$$h_{1,2} = -\frac{f(n_1 - n_0)d}{R_{2,1}n_1} \quad (16)$$

I parametri sono definiti in accordo ad Eq.(15).

I punti precedenti sono stati introdotti perchè rappresentano delle proprietà utili alla caratterizzazione dei sistemi ottici. Effettuiamo quindi il seguente studio, proposto da [3] (P.247), per verificare che i punti principali ed i punti focali posseggono effettivamente le proprietà precedentemente descritte.

Si inizia verificando la proprietà dei piani principali di definire le zone di effettiva rifrazione della lente spessa. Considerato il sistema rappresentato in Fig.(8) verrà quindi usata l'ottica matriciale al fine di ottenere la relazione in Eq. (16) per i punti principali. In pratica si vuole dimostrare che la distanza focale varia unicamente in funzione della posizione dei piani principali, quindi l'unica zona della lente che incide sulla traiettoria dei raggi è quella definita dai suddetti piani. La definizione si può riassumere nella sua interezza dimostrando che il sistema riportato in Fig. (8) focalizza nella posizione d_2 le immagini di un oggetto posto alla distanza d_1 se e solo se la distanza focale dipende dalla posizione dei piani principali secondo quanto descritto in Eq.(14).

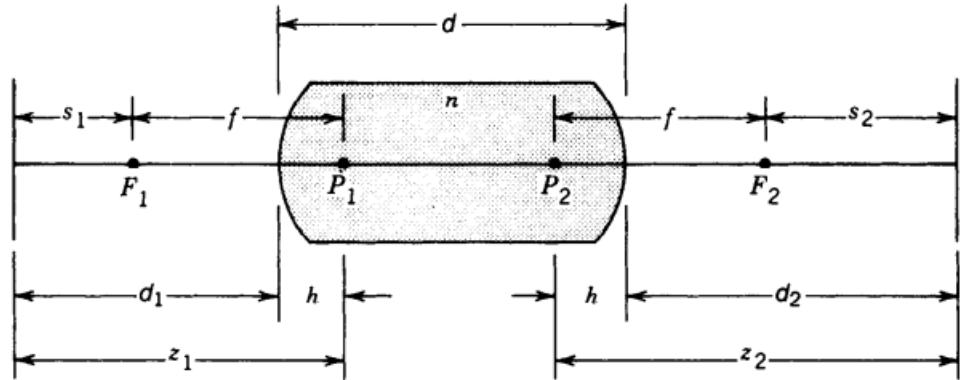


Figura 8: Schema rappresentativo dei punti cardinali di una lente spessa

- Per calcolare le posizioni dei punti si è inizialmente descritta la lente spessa rappresentata in Fig.(8), applicando gli strumenti dell'ottica matriciale. La matrice di Eq.(6) è stata espressa in funzione della distanza focale $\frac{1}{f'} = \frac{(n-1)}{R}$ ⁴.

⁴ f' , equivale alla distanza focale di una lente sottile avente raggio di curvatura R

È stata così ottenuta una nuova espressione per la matrice di rifrazione attraverso una superficie sferica:

$$\begin{bmatrix} 1 & 0 \\ \frac{(n-1)}{nR} & \frac{1}{n} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{1}{nf'} & \frac{1}{n} \end{bmatrix} \quad (17)$$

Ripetendo la procedura descritta al Cap.2.1, è stata ottenuta la matrice completa per la lente spessa riportata sopra.

$$M_{ls} = \begin{bmatrix} 1 - \frac{d}{nf'} & \frac{d}{n} \\ -\frac{(1 - \frac{d}{nf'})}{f'} - \frac{1}{f'} & -\frac{d}{nf'} + 1 \end{bmatrix}$$

Volendo descrivere le caratteristiche dell'intero sistema in Fig.(8), risulta necessario moltiplicare M_{ls} per le matrici di trasferimento in Eq. (3) nello spazio anteriore (d_1) e posteriore (d_2) alla lente. Questo passaggio è fondamentale per definire le posizioni a cui la lente focalizza oggetti posti a delle distanze d_i qualsiasi.

$$M_{tot} = \begin{bmatrix} 1 & d_2 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 - \frac{d}{nf'} & \frac{d}{n} \\ -\frac{(1 - \frac{d}{nf'})}{f'} - \frac{1}{f'} & -\frac{d}{nf'} + 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & d_1 \\ 0 & 1 \end{bmatrix} \quad (18)$$

Si è così ottenuto un sistema capace di focalizzare correttamente in una posizione d_2 l'immagine di un oggetto posto a distanza d_1 . Ciò significa che, per qualsiasi raggio entrante ad altezza y_i , l'altezza y_f del raggio finale viene definita in modo univoco dalla posizione in cui viene focalizzato l'oggetto. Quindi il valore di y_f dipende unicamente da y_i ed il valore dell'angolo di entrata non incide sull'altezza finale. Si procede studiando questa analisi teorica di un sistema matriciale usando i calcoli tipici del metodo matriciale. Affinchè la matrice risultante faccia convergere in modo definito l'immagine di un oggetto posto a distanza d_1 , è quindi necessario che nell'equazione $y_f = Ay_i + B\theta_i$ il termine B si annulli. Dunque si intende comprendere per quale valore delle variabili del sistema il termine B si può annullare.

$$B = d_1 \left(1 - \frac{d}{nf'} \right) + \frac{d}{n} + d_2 \left(d_1 \left(-\frac{2}{f'} + \frac{d}{nf'^2} \right) + \left(1 - \frac{d}{nf'} \right) \right) = 0 \quad (19)$$

Per verificare tale relazione si procede applicando le definizioni fornite per la distanza focale $\frac{1}{f}$ (Eq.(15)) e per la posizione dei punti principali h (Eq.(16)), ottenendo:

$$(d_1 + d_2) \left(1 - \frac{h}{f} \right) - \frac{d_1 d_2}{f} + \frac{d}{n} = 0 \quad (20)$$

Riscrivendo l'equazione trovata in funzione di z_1 , z_2 e sostituendo $d_{1,2} = z_{1,2} - h$ si ottiene:

$$z_1 + z_2 - \frac{z_1 z_2}{f} + b = 0 \quad (21)$$

Dove b riassume:

$$b = \frac{d}{n} - \frac{h^2}{f} - 2h\left(1 - \frac{h}{f}\right) = \frac{d}{n} + \frac{h}{f}(h - 2f) \quad (22)$$

Si può notare che la relazione in Eq.(21) risulta verificata se il termine b si annulla:

$$b = 0 \Rightarrow \frac{1}{z_1} + \frac{1}{z_2} = \frac{1}{f}$$

Per mostrare che $b = 0$ applichiamo nuovamente le definizioni di distanza focale Eq.(15) e di posizione dei piani principali Eq.(16), ottenendo:

$$\begin{aligned} b &= \frac{d}{n} - h \frac{f'}{f} \\ &\Downarrow \\ b &= h \frac{f'}{f} - h \frac{f'}{f} = 0 \quad \square \end{aligned}$$

- In precedenza sono stati introdotti i punti focali e la distanza focale. Procediamo ora a dimostrare la proprietà dei punti focali di far convergere in un determinato punto F un qualsiasi raggio parallelo rispetto all'asse ottico. Verifichiamo quindi che dato un raggio incidente ad un'altezza y_1 sulla prima superficie, questo passerà sempre per il punto F_2 posto a distanza $f - h$ rispetto alla seconda superficie della lente.

La matrice, per il sistema in esame è la seguente:

$$M_{tot} = \begin{bmatrix} 1 & f - h \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 - \frac{d}{nf'} & \frac{d}{n} \\ -\frac{(1 - \frac{d}{nf'})}{f'} - \frac{1}{f'} & 1 - \frac{d}{nf'} \end{bmatrix} \quad (23)$$

L'obiettivo è dimostrare che F_2 è un punto focale del sistema. Questo è verificato se si dimostra che per raggi paralleli entranti ($\theta_1 = 0$), il raggio interseca l'asse ottico ($y_2 = 0$), nella posizione corrispondente a F_2 ($d = f - h$).

Considerando i risultati di Eq.(9): $y_2 = Ay_1 + B\theta_1$, si nota che per $A = 0$ si ottiene $y_2 = B\theta_1$. Ne consegue che y_2 si annulla nel caso di raggi entranti, paralleli all'asse ottico, $\theta_1 = 0$, ossia il caso che si sta considerando. Per dimostrare che $y_2 = 0$ rimane quindi da verificare che il termine A trovato è nullo. Considerata l'equazione che definisce A :

$$A = \left(1 - \frac{d}{nf'}\right) + (f - h)\left(-\frac{2}{f'} + \frac{d}{nf'^2}\right) \quad (24)$$

applicando la definizione di distanza focale (15) ed usando la relazione $2f - h = f'$ si ottiene:

$$A = \left(1 - \frac{h}{f}\right) + \frac{(f - h)}{-f} = 0 \quad \square \quad (25)$$

2.2.2 Esempio per una lente spessa

L'analisi viene proseguita considerando un caso specifico di una lente spessa, calcolandone la posizione dei punti focali e dei piani principali. Questi calcoli vengono effettuati riscrivendo l'espressione di distanza focale Eq.(15) e posizione dei punti principali Eq.(16), usando gli strumenti dell'ottica matriciale.

Per ottenere la distanza focale, si considera quanto mostrato in Fig.(9). Le coordinate di ingresso del raggio sono $(y_o, 0)$, e quelle di uscita sono (y_f, θ_f) . Dunque l'equazione del raggio per questo sistema è:

$$\begin{aligned} y_f &= Ay_0 \\ \theta_f &= Cy_0 \end{aligned} \quad (26)$$

Si osserva in Fig.(9) che per piccoli angoli vale:

$$\theta_f = -\frac{y_f}{q_2} \quad (27)$$

$$\Rightarrow q_2 = -\frac{y_f}{\theta_f} = -\frac{A}{C} \quad \square \quad (28)$$

In maniera simile ad Eq.(27), si può esprimere $\theta_f = -\frac{y_0}{f_2}$ e dunque:

$$f_2 = -\frac{y_0}{\theta_f} = -\frac{1}{C} \quad \square \quad (29)$$

Come si può vedere il risultato è coerente con la descrizione matriciali per le lenti sottili, Eq.(7).

In fine utilizzando Eq. (28) e (29) è possibile esprimere la distanza h_2 :

$$h_2 = q_2 - f_2 = -\frac{A}{C} + \frac{1}{C} = \frac{1-A}{C} \quad \square \quad (30)$$

Proseguendo similmente è possibile ottenere le restanti posizioni dei punti focali e principali:

$$q_1 = \frac{D}{C} ; \quad h_1 = \frac{D - n_0/n_f}{C} ; \quad f_1 = q_1 - h_1 = \frac{n_0/n_f}{C} \quad (31)$$

Per una trattazione completa consultare [4].

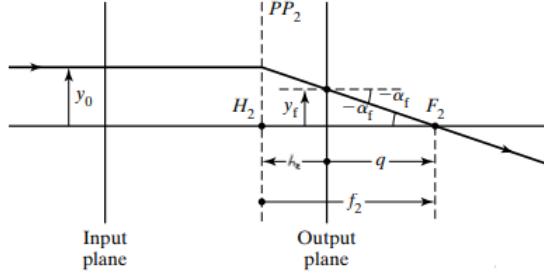


Figura 9: Schema della seconda superficie di una lente spessa

Si completa la analisi per una lente spessa con le seguenti caratteristiche:

	$R_1(\text{mm})$	$R_2(\text{mm})$	n_{vetro}	spessore(mm)
Lente A	92.51	-148.70	1.6110	50.83

Sviluppando la matrice in Eq.(12) si ottiene:

$$M_{ls} = \begin{bmatrix} 0.87 & 19.58 \\ -0.01 & 0.91 \end{bmatrix}$$

Applicando le formule ricavate in precedenza: Eq.(29), Eq.(30) si ottengono i seguenti valori per la distanza focale e posizione dei punti principali rispetto ai vertici della lente:

$$f = -98.22 \text{ mm} ; \quad h_1 = 7.90 \text{ mm} ; \quad h_2 = -12.70 \text{ mm} \quad (32)$$

Si potrà vedere che i valori ottenuti per la distanza focale e la posizione dei piani principali coincidono con i risultati del prossimo sistema ottico che verrà trattato (Cap.(2.3.1)). Questa uguaglianza implica che i due sistemi ottici che stiamo considerando focalizzano la luce alla stessa distanza rispetto ai piani principali della lente. I due sistemi saranno dunque equivalenti per quanto riguarda la posizione di raggi focalizzati.

2.3 Applicazione del metodo ai sistemi composti

L'introduzione dell'approccio matriciale ha permesso di definire delle componenti modulari, ideali per la descrizione di sistemi composti di elevata complessità. Si può quindi applicare quanto imparato rappresentando matematicamente il tripletto di Cooke. Così facendo sarà possibile ottenere delle informazioni sulle posizioni dei punti cardinali precedentemente introdotti.

Il caso che si sta considerando è quello di un sistema composto, a cascata, la cui descrizione matriciale può essere affrontata a partire dal prodotto degli elementi che costituiscono l'apparato completo. Vediamo in Fig.(10) gli elementi di cui si compone il tripletto:

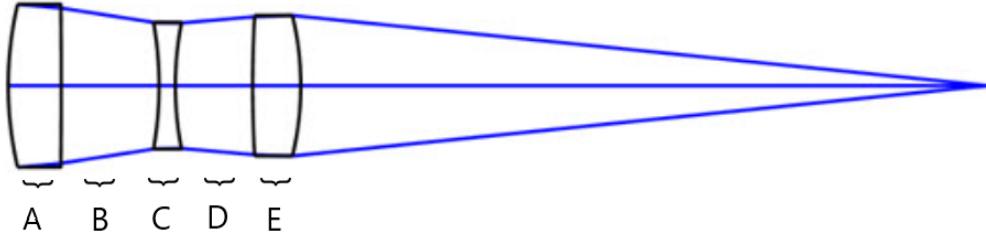


Figura 10: Componenti del tripletto di Cooke.

Il sistema è evidentemente divisibile in due categorie, le tre lenti spesse e i due spazi d'aria fra le lenti. Avendo già calcolato le espressioni matriciali di questi elementi, rappresentare matricialmente il tripletto di Cooke è un semplice gioco di moltiplicazione delle componenti. Rifacendosi allo schema di riportato in Fig(10) si definiscono le cinque matrici che identificano il tripletto.

Gli elementi A, C, E sono delle lenti spesse, la cui matrice è descritta da Eq.(12), mentre B, D identificano degli spazi di aria fra le lenti Eq.(3). Si calcola dunque il prodotto fra matrici secondo l'ordine citato, nella trattazione, a Cap.2.1, ottenendo così la rappresentazione matriciale del tripletto di Cooke.

$$M_{tc} = M_e \cdot M_d \cdot M_c \cdot M_b \cdot M_a$$

$$M_{tc} = \begin{bmatrix} 1 + \frac{d_e(n_0-n_1)}{n_1^2 R_{e1}} & \frac{d_e n_0}{n_1^2} \\ \frac{(n_0-n_1)\left(\frac{n_1}{n_0} + \frac{d_e(n_1-n_0)}{n_0 n_1 R_{e2}}\right)}{n_1 R_{e1}} + \frac{(n_1-n_0)}{n_0 R_{e2}} & \frac{n_0\left(\frac{n_1}{n_0} + \frac{d_e(n_1-n_0)}{n_0 n_1 R_{e2}}\right)}{n_1} \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{d_d}{n_0} \\ 0 & 1 \end{bmatrix}.$$

$$\begin{bmatrix} 1 + \frac{d_c(n_0-n_2)}{n_2^2 R_{c1}} & \frac{d_c n_0}{n_2^2} \\ \frac{(n_0-n_2)\left(\frac{n_2}{n_0} + \frac{d_c(n_2-n_0)}{n_0 n_2 R_{c2}}\right)}{n_2 R_{c1}} + \frac{(n_2-n_0)}{n_0 R_{c2}} & \frac{n_0\left(\frac{n_2}{n_0} + \frac{d_c(n_2-n_0)}{n_0 n_2 R_{c2}}\right)}{n_2} \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{d_b}{n_0} \\ 0 & 1 \end{bmatrix}.$$

$$\begin{bmatrix} 1 + \frac{d_a(n_0-n_1)}{n_1^2 R_{a1}} & \frac{d_a n_0}{n_1^2} \\ \frac{(n_0-n_1)\left(\frac{n_1}{n_0} + \frac{d_a(n_1-n_0)}{n_0 n_1 R_{a2}}\right)}{n_1 R_{a1}} + \frac{(n_1-n_0)}{n_0 R_{a2}} & \frac{n_0\left(\frac{n_1}{n_0} + \frac{d_a(n_1-n_0)}{n_0 n_1 R_{a2}}\right)}{n_1} \end{bmatrix}$$

Per svolgere i calcoli e visualizzare le matrici è stato sfruttato il software Wolfram Alpha [13], in Fig.(11) è possibile osservare il codice utilizzato.

```
"Scrittura delle matrici utilizzate:"
" Lente spessa A:"
MA = {{1, 0}, {-(n0 - n1) / (n0 * Ra2), n1 / n0}}.{{1, da / n1}, {0, 1}}.{{1, 0}, {-(n1 - n0) / (n1 * Ra1), n0 / n1}}
"Propagazione in aria, spazio B:"
MB = {{1, db / n0}, {0, 1}}
" Lente spessa C:"
MC = {{1, 0}, {-(n0 - n2) / (n0 * Rc2), n2 / n0}}.{{1, dc / n2}, {0, 1}}.{{1, 0}, {-(n2 - n0) / (n2 * Rc1), n0 / n2}}
"Propagazione in aria, spazio D:"
MD = {{1, dd / n0}, {0, 1}}
" Lente spessa E:"
ME = {{1, 0}, {-(n0 - n1) / (n0 * Re2), n1 / n0}}.{{1, de / n1}, {0, 1}}.{{1, 0}, {-(n1 - n0) / (n1 * Re1), n0 / n1}}
" Moltiplicazione fra le precedenti matrici:"
TC = ME.MD.MC.MB.MA // MatrixForm
```

Figura 11: Codice utilizzato il calcolo della matrice del tripletto di Cooke.

Si vede che ad ogni espressione corrisponde una matrice. Una volta definite tutte le matrici in accordo a quanto descritto in precedenza, si effettua una moltiplicazione matriciale mediante l'apposito comando di Wolfram Alpha. Date le dimensioni molto ingombranti non è possibile riportare il risultato ottenuto per la matrice del tripletto di Cooke.

2.3.1 Proprietà del tripletto di Cooke

Replicando quanto fatto per la lente spessa si identificherà un caso specifico per il tripletto di Cooke, determinandone poi la distanza focale e la posizione dei piani principali. Si considerano dunque i seguenti valori, riportati nelle specifiche di un particolare tripletto di Cooke descritto dal libro [4].

	R_1 (mm)	R_2 (mm)	n_{vetro}	spessore(mm)
Lente A	19.4	-128.3	1.6110	4.29
Lente C	-57.8	18.9	1.5744	0.93
Lente E	311.3	-66.4	1.6110	3.03

Per gli spazi di aria si usa: $d_b = 1.63$ mm, $d_d = 12.90$ mm.

La matrice complessiva è stata calcolata sfruttando Wolfram Alpha, per effettuare i calcoli sarà quindi sufficiente sostituire i valori numerici all'interno delle espressioni precedentemente ottenute. In Fig.(12) è possibile vedere i valori sostituiti e la sintassi utilizzata all'interno del programma.

```

"Sostituzione dei valori all'interno delle precedenti matrici:"
"Lente spessa A:"
MA /. {n0 → 1, n1 → 1.6110, Ra1 → 19.4, Ra2 → -128.3, da → 4.29}
"Propagazione in aria, spazio B:"
MB /. {n0 → 1, db → 1.63}
"Lente spessa C:"
MC /. {n0 → 1, n2 → 1.5744, Rc1 → -57.8, Rc2 → 18.90, dc → 0.93}
"Propagazione in aria, spazio D:"
MD /. {n0 → 1, dd → 12.90}
"Lente spessa E:"
ME /. {n0 → 1, n1 → 1.6110, Re1 → 311.3, Re2 → -66.4, de → 3.03}

```

Figura 12: Assegnazione dei valori alle variabili delle matrici

Si ottiene così la seguente moltiplicazione di matrici:

$$M_{tc} = \underbrace{\begin{bmatrix} 0.95 & 1.65 \\ -0.03 & 0.99 \end{bmatrix}}_E \cdot \underbrace{\begin{bmatrix} 1 & 1.63 \\ 0 & 1 \end{bmatrix}}_D \cdot \underbrace{\begin{bmatrix} 1.00 & 0.37 \\ 0.04 & 1.01 \end{bmatrix}}_C \cdot \underbrace{\begin{bmatrix} 1 & 12.9 \\ 0 & 1 \end{bmatrix}}_B \cdot \underbrace{\begin{bmatrix} 0.99 & 1.16 \\ -0.01 & 0.98 \end{bmatrix}}_A \quad (33)$$

Sempre applicando le funzionalità della piattaforma online è stato svolto il prodotto fra matrici in Eq.(38), ottendendo come risultato la seguente matrice 2×2 :

$$M_{tc} = \begin{bmatrix} 0.8706 & 19.58 \\ -0.01018 & 0.9195 \end{bmatrix} \quad (34)$$

In alternativa al calcolo precedentemente mostrato è possibile sostituire direttamente i valori numerici all'interno della matrice completa ottenuta da Fig.(12). Questo metodo di sostituzione, rispetto a quello precedentemente considerato, è sia più veloce che più efficace per sostituire le variabili all'interno del sistema. In Fig.(13) viene riportata la sostituzione da effettuare nel caso appena descritto.

```

"Ottenimento della matrice del tripelto di Cooke dalla moltiplicazione delle matrici delle singoli componenti:"
TC = ME.MD.MC.MB.MA
"Sostituzione dei valori numerici all'interno della matrice totale:"
TC /. {n0 → 1, n1 → 1.6110, n2 → 1.5744, da → 4.29, db → 1.63, dc → 0.93, dd → 12.90, de → 3.03,
Ra1 → 19.4, Ra2 → -128.3, Rc1 → -57.8, Rc2 → 18.9, Re1 → 311.3, Re2 → -66.4}

```

Figura 13: Assegnazione dei valori alla matrice del tripelto di Cooke

Sfruttando questi risultati e considerando quanto ottenuto in Eq.(34), Eq.(29), Eq.(30) si ottengono i dati cercati di lunghezza focale f e posizione dei piani principali $h_{1,2}$ rispetto ai vertici della lente.

$$f = -98.23 \text{ mm} ; h_1 = 7.90 \text{ mm} ; h_2 = -12.71 \text{ mm} \quad (35)$$

2.4 Aberrazioni cromatiche e metodo matriciale per raggi parassiali

Nel contesto delle aberrazioni cromatiche, il metodo matriciale può essere adattato per includere gli effetti della dispersione cromatica. Poiché le aberrazioni cromatiche derivano dal fatto che il potere rifrattivo di una lente varia con la lunghezza d'onda, il metodo matriciale per raggi parassiali può essere applicato separatamente a ciascuna lunghezza d'onda di interesse. Si calcolano di seguito le matrici ABCD corrispondenti a lunghezze d'onda della luce tipicamente usate per evidenziare il fenomeno delle aberrazioni cromatiche: rosso, verde e blu. Sostituendo i rispettivi valori di indice di rifrazione per ciascuna lunghezza d'onda della luce si otterranno matrici differenti per ciascun caso. I dati relativi agli indici di rifrazione del vetro sono stati ottenuti dalla base di dati disponibile online (Refractive Index Database)[14], in particolare si è considerato il vetro N-SK2 nelle varianti "dense Crown" e "light Flint"⁵. Si sostituiscono quindi i seguenti valori per l'indice di rifrazione dei vetri per luce di lunghezza d'onda: rossa 740 nm, verde 533 nm e blu 485 nm.

$\lambda(\text{nm})$	740	533	485
n_c	1.6013	1.6109	1.6150

Una volta determinati gli indici di rifrazione delle lenti si procede implementando i valori corretti all'interno della espressione della lente spessa già ottenuta in Eq.(12) utilizzando Wolfram Alpha [13].

```
"Definizione delle matrici per lenti spesse e sostituzione dei valori relativi a ciascuna lunghezza d'onda:"
"Luci rosse:"
MR = {{1, 0}, {-(n0 - n1)/(n0 * Ra2), n1/n0}}.{{1, da/n1}, {0, 1}}.{{1, 0}, {-(n1 - n0)/(n1 * Ra1), n0/n1}}
MR /. {n0 → 1, n1 → 1.6013, Ra1 → 92.51, Ra2 → -148.70, da → 50.83}
"Luci verdi:"
MV = {{1, 0}, {-(n0 - n1)/(n0 * Ra2), n1/n0}}.{{1, da/n1}, {0, 1}}.{{1, 0}, {-(n1 - n0)/(n1 * Ra1), n0/n1}}
MV /. {n0 → 1, n1 → 1.6109, Ra1 → 92.51, Ra2 → -148.70, da → 50.83}
"Luci blu:"
MB = {{1, 0}, {-(n0 - n1)/(n0 * Ra2), n1/n0}}.{{1, da/n1}, {0, 1}}.{{1, 0}, {-(n1 - n0)/(n1 * Ra1), n0/n1}}
MB /. {n0 → 1, n1 → 1.6150, Ra1 → 92.51, Ra2 → -148.70, da → 50.83}
```

Figura 14: Sostituzione dei valori numerici all'interno delle matrici per diverse lunghezze d'onda della luce.

Si ottengono così le seguenti matrici:

$$M_{rosso} = \begin{bmatrix} 0.87 & 19.82 \\ -0.01 & 0.92 \end{bmatrix} \quad M_{verde} = \begin{bmatrix} 0.87 & 19.58 \\ -0.01 & 0.91 \end{bmatrix} \quad M_{blu} = \begin{bmatrix} 0.87 & 19.48 \\ -0.01 & 0.91 \end{bmatrix} \quad (36)$$

Come è possibile vedere le matrici hanno elementi con valori numerici molto simili fra loro. Infatti il cammino dei raggi sarà pressoché lo stesso e le differenze saranno evidenziabili solo attraverso analisi più dettagliate.

⁵Indicati con gli indici: n_c , n_f

Ripetendo le analisi effettuate a Cap.(2.3.1) e Cap.(2.2.2), considerando quanto ottenuto in Eq.(34), Eq.(29), Eq.(30) si ottengono i dati di lunghezza focale f e posizione dei piani principali $h_{1,2}$ per le tre diverse lunghezze d'onda della luce.

$$\begin{aligned} f_{rosso} &= -104.64 \text{ mm} ; \quad h_1 = 7.99 \text{ mm} ; \quad h_2 = 12.55 \text{ mm} \\ f_{verde} &= -104.61 \text{ mm} ; \quad h_1 = 7.90 \text{ mm} ; \quad h_2 = 12.42 \text{ mm} \\ f_{blu} &= -104.68 \text{ mm} ; \quad h_1 = 7.86 \text{ mm} ; \quad h_2 = 12.36 \text{ mm} \end{aligned} \quad (37)$$

Si effettua in seguito la stessa analisi per il caso del tripletto di Cooke. Considerando le stesse lunghezze d'onda trattate in precedenza si considerano i seguenti valori per gli indici di rifrazione:

$\lambda(\text{nm})$	740	533	485
n_c	1.6013	1.6109	1.6150
n_f	1.5736	1.5861	1.5916

Si procede, quindi, sostituendo i valori corretti all'interno della espressione già ottenuta a (Cap.2.3.1) per il tripletto di Cooke.

```
"Definizione della matrice per il tripletto di Cooke"
TC = ME.MD.MC.MB.MA // MatrixForm
"Sostituzione dei valori relativi a ciascuna lunghezza d'onda:"
"rosso:"
TC /. {n0 → 1, n1 → 1.6013, n2 → 1.5736, da → 4.29, db → 1.63, dc → 0.93, dd → 12.90, de → 3.03,
Ra1 → 19.4, Ra2 → -128.3, Rc1 → -57.8, Rc2 → 18.9, Re1 → 311.3, Re2 → -66.4}
"verde:"
TC /. {n0 → 1, n1 → 1.6109, n2 → 1.5861, da → 4.29, db → 1.63, dc → 0.93, dd → 12.90, de → 3.03,
Ra1 → 19.4, Ra2 → -128.3, Rc1 → -57.8, Rc2 → 18.9, Re1 → 311.3, Re2 → -66.4}
"blu"
TC /. {n0 → 1, n1 → 1.6150, n2 → 1.5916, da → 4.29, db → 1.63, dc → 0.93, dd → 12.90, de → 3.03,
Ra1 → 19.4, Ra2 → -128.3, Rc1 → -57.8, Rc2 → 18.9, Re1 → 311.3, Re2 → -66.4}
```

Figura 15: Sostituzione dei valori numerici all'interno delle matrici per diverse lunghezze d'onda della luce.

Ottenendo in definitiva le seguenti matrici:

$$M_{rosso} = \begin{bmatrix} 0.88 & 19.63 \\ -0.009 & 0.92 \end{bmatrix} \quad M_{verde} = \begin{bmatrix} 0.88 & 19.62 \\ -0.009 & 0.92 \end{bmatrix} \quad M_{blu} = \begin{bmatrix} 0.88 & 19.61 \\ -0.009 & 0.92 \end{bmatrix} \quad (38)$$

Si conclude la trattazione ricavando i dati di lunghezza focale f e posizione dei piani principali $h_{1,2}$ per le tre diverse lunghezze d'onda della luce.

$$\begin{aligned}
f_{rosso} &= -104.64 \text{ mm} ; \quad h_1 = 8.04 \text{ mm} ; \quad h_2 = 12.55 \text{ mm} \\
f_{verde} &= -104.61 \text{ mm} ; \quad h_1 = 8.16 \text{ mm} ; \quad h_2 = 12.42 \text{ mm} \\
f_{blu} &= -104.68 \text{ mm} ; \quad h_1 = 8.22 \text{ mm} ; \quad h_2 = 12.36 \text{ mm}
\end{aligned} \tag{39}$$

3 Il metodo del ray tracing

Il metodo del ray tracing è una tecnica di simulazione utilizzata per modellare il comportamento della luce visibile in un sistema ottico. Questo metodo trova molteplici applicazioni. Campi come l'ottica geometrica, la grafica computerizzata, l'acustica e le simulazioni radar, trovano nel ray tracing il modello ideale per descrizioni accurate. È un metodo computazionale, basato sull'applicazione di formule matematiche, molto utile nello studio di precisione di sistemi composti come il tripletto di Cooke.

Il ray tracing sfrutta i principi dell'ottica geometrica per tracciare il percorso della luce, e non necessita di altre approssimazioni (es. raggi parassiali). In questo modo si riesce a rappresentare sistemi reali senza mascherarne le imperfezioni pratiche. Ciò che si ottiene è quindi la rappresentazione più fedele al percorso effettivo compiuto dai singoli raggi che si possa ottenere sfruttando i presupposti dell'ottica geometrica [4].

3.1 Descrizione generale del metodo

Il ray tracing si basa sulla nozione di raggi ottici, caratteristica dell'ottica geometrica. Nell'approssimazione dei raggi ottici la luce si propaga in linea retta fino a quando non incontra una superficie o viene immersa in particolari mezzi ottici. In questi casi, il percorso del raggio subisce mutazioni a causa di fenomeni quali riflessione, rifrazione, o diffusione.

Procediamo dunque ad analizzare i passaggi che caratterizzano il metodo.

- Come prima cosa è necessario generare i raggi ottici per la sorgente luminosa che si ha intenzione di studiare. Questi raggi possono essere emessi direttamente da componenti luminosi come le lampadine o provenire da riflessioni su superfici illuminate.
- Si procede dunque a tracciare il percorso di ciascun singolo raggio calcolando tutte le interazioni con mezzi e superfici. I fenomeni descritti comprendono la riflessione, la rifrazione ed i fenomeni di assorbimento o dispersione. Nel caso della riflessione il raggio viene rimandato da una superficie secondo quanto descritto dalle leggi della riflessione: l'angolo di incidenza è uguale all'angolo riflesso. La rifrazione definisce il comportamento di un raggio che passa da un mezzo a un altro (passaggio aria-vetro), questo raggio subisce una deviazione angolare che viene descritta dalla legge di Snell. L'assorbimento o dispersione riassumono la proprietà di alcuni materiali di assorbindre parte della luce del raggio entrante o disperderla in più direzioni. L'effetto descritto è tipico delle superfici opache o ruvide, la cui struttura microscopica determina riflessioni in più direzioni. Questa tipologia di superfici viene detta

superficie Lambertiana. Le superfici lambertiane sono superfici per cui l'intensità luminosa riflessa diminuisce proporzionalmente al coseno dell'angolo di osservazione $I(\theta) = I_0 \cos(\theta)$ (Legge di Lambert).

- Un altro passaggio fondamentale è la determinazione del punto di intersezione dei raggi ottici con le superfici e gli oggetti. Le intersezioni sono calcolate matematicamente, considerando la geometria degli oggetti (sfere, piani, poligoni, ecc.).
- Per una descrizione completa è necessario trattare anche i raggi secondari. Dopo un'interazione, possono essere generati raggi secondari che continuano a viaggiare nel sistema. Il tracciamento di questi raggi non costituisce una priorità e viene eseguito fino a quando i raggi non escono dal campo di interesse o vengono assorbiti.
- Il processo di tracciamento dei raggi si conclude con il calcolo dell'intensità luminosa. Ogni raggio trasporta un certo ammontare di energia luminosa. L'intensità della luce che arriva su un punto finale viene calcolata in base al numero di raggi incidenti, all'energia trasportata dai raggi e alle interazioni fra i raggi incidenti e la superficie finale.

Fra le applicazioni principali di questo metodo vi è l'ottica geometrica. Il ray tracing è essenziale per modellare sistemi ottici più complessi come lenti, specchi e strumenti ottici composti. L'utilizzo principale è nel calcolo del percorso della luce in sistemi composti e nell'analisi delle aberrazioni ottiche. In particolar modo sfruttando il ray tracing è possibile eliminare le aberrazioni ottiche introdotte da sistemi complessi di lenti. Questo metodo viene applicato anche nella grafica computerizzata e l'acustica. Nella grafica 3D, il ray tracing viene impiegato per generare immagini altamente realistiche. Il metodo permette di simulare il comportamento della luce in ambienti complessi, descrivendo le ombre, i fenomeni di riflessione, e le rifrazioni su materiali trasparenti o lucidi. In acustica, il ray tracing è usato per modellare la propagazione del suono in spazi complessi, come teatri o arene. I raggi sonori possono essere trattati usando i principi dell'ottica geometrica, consentendo di effettuare studi per l'ottimizzazione acustica di un ambiente.

I vantaggi del metodo sono diversi. L'aspetto principale è l'accuratezza, poichè il metodo non ha altre approssimazioni fuori dalle definizioni dell'ottica geometrica, riesce a fornire simulazioni molto precise di come la luce si comporta in un sistema. Un altro vantaggio è la capacità di rendere possibile la rappresentazione di fenomeni complessi, molto difficili da descrivere con altri mezzi. Fra gli svantaggi spicca la complessità computazionale. Il calcolo del percorso dei raggi attraverso interazioni multiple può richiedere un notevole sforzo computazionale. Questo è particolarmente vero quando si tracciano decine di migliaia di raggi in simulazioni grafiche o ottiche dettagliate, rendendo quindi necessaria l'applicazione di tecniche avanzate di ottimizzazione.

L'applicazione del metodo che verrà trattata in questa tesi è volta alla definizione del percorso dei raggi luminosi in un sistema composto di lenti. Si procede a fornire una descrizione di alcune delle operazioni precedentemente introdotte: la propagazione in un mezzo e la rifrazione sulle superfici.

- Per definire la propagazione di una retta $y - y_0 = m(x - x_0)$ è sufficiente individuare il coefficiente angolare ed applicare l'equazione. Nel caso di raggi luminosi che si propagano

in un mezzo di indice di rifrazione n , il coefficiente angolare è dato dal rapporto fra n e la tangente dell'angolo fra la retta e l'orizzontale, θ . Nel caso di sistemi di lenti l'asse orizzontale viene individuato dall'asse ottico.

$$m = \frac{\tan \theta}{n} \quad (40)$$

- La rifrazione su una superficie determina un cambiamento nell'angolo di propagazione in accordo con la legge di Snell.

$$n_1 \sin (\theta'_1) = n_2 \sin (\theta'_2) \quad (41)$$

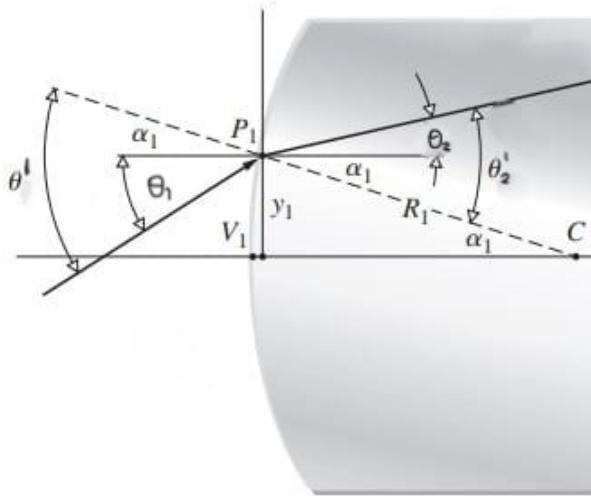


Figura 16: Rifrazione su una superficie sferica

Risulta importante specificare che l'angolo θ'_i , è diverso dell'angolo θ_i a cui si sta propagando il raggio. Come è possibile vedere in Fig.(16) θ'_i rappresenta l'angolo fra il raggio incidente e la normale alla superficie considerata. Nel caso particolare di una superficie sferica, θ'_i è dato dalla somma fra l'angolo θ_i e l'angolazione della superficie sferica nel punto di contatto α . Sfruttando della semplice trigonometria si dimostra che l'angolo della superficie sferica di raggio R ad un'altezza y rispetto all'asse ottico è dato da:

$$R \sin (\alpha) = y \Rightarrow \alpha = \arcsin \left(\frac{y}{R} \right)$$

È quindi possibile ricavare l'angolo di rifrazione θ_2 da (41):

$$\theta_2 = \arcsin \left(\frac{n_1}{n_2} \cdot \sin(\theta_1 + \alpha) \right) - \alpha \quad (42)$$

Una volta individuate, le formule matematiche (41), (42) sono state applicate nella scrittura di programmi di ray tracing su Python [5].

3.2 Applicazione del metodo per una lente spessa

Si è inizialmente proceduto a scrivere il programma di ray tracing per il caso più semplice di una lente spessa. Lo scopo del programma è ottenere un grafico che permetta di evidenziare i punti di intersezione dei raggi con l'asse ottico. Gli step logici seguiti dal programma vengono riassunti dallo schema a blocchi in Fig. (17).

Si osserva che il programma (App. A) è composto da tre sezioni principali.

1. Definizione iniziale delle funzioni e dei parametri:

In questa sezione iniziale del programma vengono definite le variabili ed i parametri del programma. Le funzioni necessarie per calcolare il percorso dei raggi vengono scritte sotto forma di funzioni richiamabili, affinché sia possibile applicarle più facilmente in altre parti nel programma. Vengono poi caratterizzate le lenti attraverso l'assegnazione di parametri quali lo spessore, i raggi delle superfici e gli indici di rifrazione. Infine sono definiti i vettori dei raggi, contenenti come informazioni la posizione y e l'angolo θ del raggio rispetto all'asse ottico.

2. Calcolo del percorso:

La sezione centrale del programma riguarda il calcolo del percorso dei raggi. È questa la parte in cui sono state applicate le formule in Eq.(41) ed Eq.(42), e sono state sfruttate le leggi di propagazione in un mezzo e la legge di Snell per determinare il percorso della luce. Per calcolare la propagazione è necessario conoscere l'esatto punto di intersezione fra il raggio e le lenti, al fine di determinare correttamente la distanza percorsa raggio in ogni mezzo. Quindi si ricava l'angolo rifratto in funzione dell'altezza y^i a cui ogni raggio incide sulla lente.

3. Visualizzazione dei risultati:

Una volta effettuati questi calcoli, i risultati ottenuti vengono applicati alla stima del FWHM⁶, quindi alla dispersione dei raggi sul piano focale. Le proprietà dei sistemi vengono inoltre rappresentate sotto forma di grafici. Il primo grafico riporta l'intero sistema, dalla prima lente alla posizione in cui i raggi vengono focalizzati. Il percorso di ciascun raggio viene raffigurato con un colore preciso per poter meglio comprendere il percorso effettuato. Il secondo grafico consiste in un ingrandimento della zona di intersezione fra i raggi e l'asse ottico ($y = 0$). Un altro grafico riporta la distribuzione con cui un numero arbitrario di raggi entranti incide sul piano focale mentre l'ultimo grafico serve a quantificare la dispersione delle zone di incidenza dei raggi sul piano focale con riferimento all'asse ottico.

⁶Full Width at Half Maximum

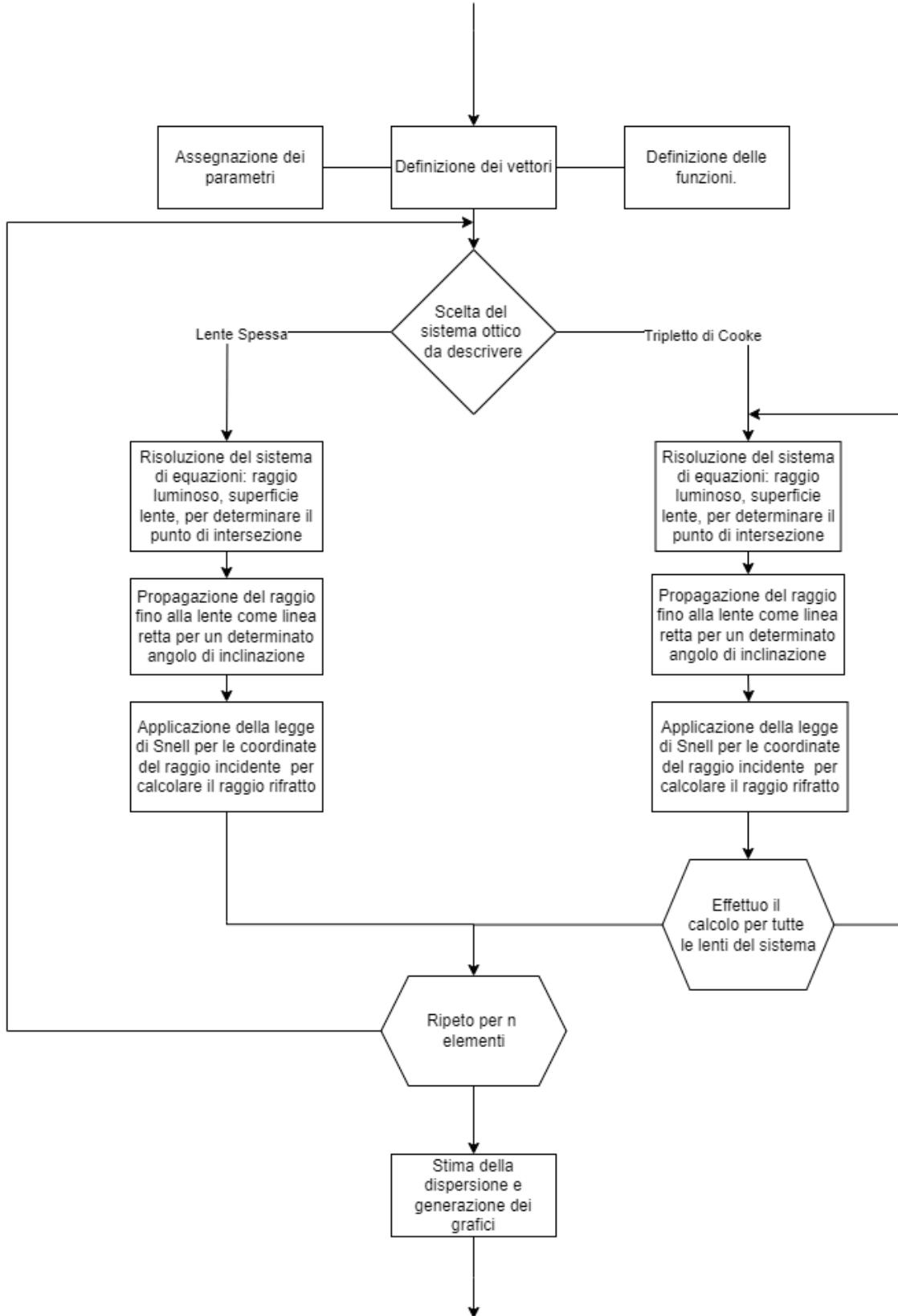


Figura 17: Diagramma a blocchi del programma di ray tracing.

Applicando il programma al caso specifico della lente spessa descritta a Cap.(2.2.2) si ottiene la seguente configurazione:

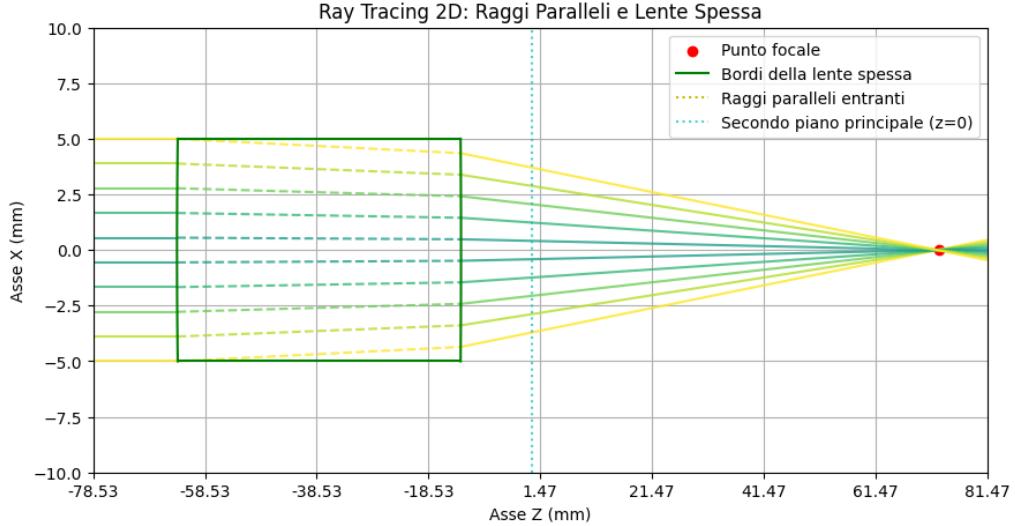


Figura 18: Ray tracing di una lente spessa.

Si noti che i diversi colori dei raggi servono ad aiutare a distinguere il percorso effettuato da ogni singolo raggio e non definiscono raggi luminosi aventi diverse lunghezze d'onda. Nel programma, la lente è stata posta ad una posizione $z_0 = 15$ mm, a partire da questa informazione ed utilizzando i risultati di Eq.(29), Eq.(28) otteniamo il punto in cui ci aspettiamo che i raggi intersechino l'asse z:

$$f_z = z_0 + dz + f - h_2 = z_0 + dz + q_2 \quad (43)$$

Applicando Eq.(28), trovo che il valore di q_2 per il sistema considerato è: 85.52 mm.

Ne consegue:

$$f_2 = 151.36 \text{ mm} \quad (44)$$

Prima di confrontare questo risultato con la rappresentazione grafica si osserva che l'origine dell'asse Z del grafico viene posta in corrispondenza alla posizione del secondo piano principale.

$$h_z = z_0 + dz + h_2 = 78.53 \text{ mm} \quad (45)$$

Si ottiene quindi la posizione del grafico f_z in cui i raggi luminosi vengono fatti convergere:

$$f_z = f_2 - h_z = 151.36 - 78.53 = 72.83 \text{ mm} \quad (46)$$

Osservando il grafico in Fig.(18) si vede come i raggi vengono correttamente fatti convergere nel punto focale calcolato al Cap.(2.3.1) dimostrando l'accordo fra il modello teorico matriciale ed il ray tracing. Studiamo ora in dettaglio la zona in cui i raggi ottici intersecano il piano focale. Ingrandendo in un intorno del punto focale risulta evidente il fenomeno delle aberrazioni sferiche. Il grafico in Fig.(19) fornisce un ingrandimento nella zona considerata. Risulta inoltre evidente

come i raggi ottici che intersecano la lente ad un'altezza maggiore rispetto all'asse ottico (raggi di colore giallo) vengono fatti convergere ad una distanza maggiore rispetto al punto focale. Questo risultato è perfettamente in accordo con l'approssimazione per raggi parassiali.

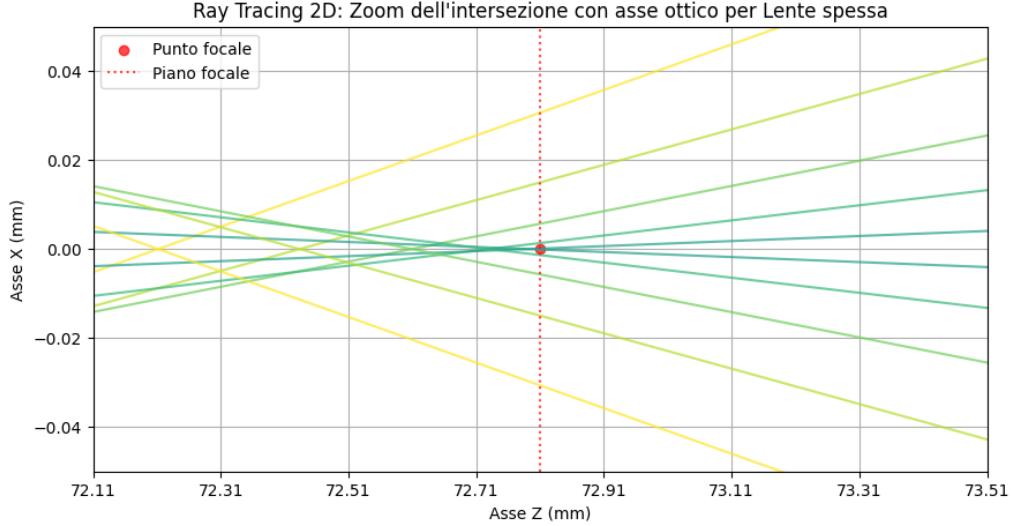


Figura 19: Ingrandimento in corrispondenza del punto focale in funzione della distanza dal secondo piano principale.

3.3 Applicazione del metodo al tripletto di Cooke

La costruzione del programma di ray tracing per il tripletto di Cooke procede ampliando il programma per una lente spessa e calcolando il percorso dovuto a due ulteriori lenti. Facendo riferimento al diagramma a blocchi in Fig.(17) si vede come la differenza rispetto al programma per una singola lente consiste nel ripetere l'operazione centrale del calcolo per un totale di tre lenti spesse invece che una. Si applica il programma per il sistema descritto a Cap.(2.3) ottenendo così la configurazione in Fig.(20)

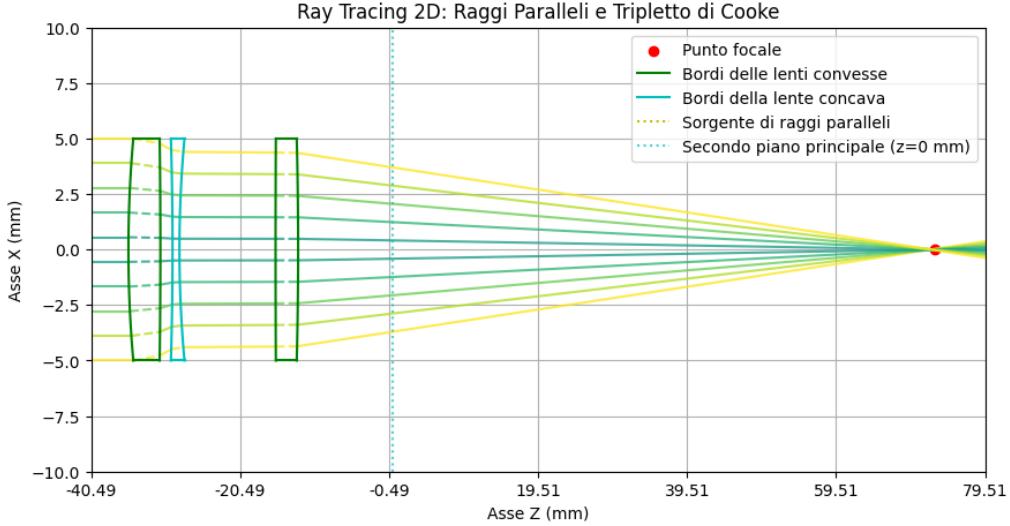


Figura 20: Ray tracing del tripletto di Cooke.

Calcoliamo ora la predizione teorica della posizione in cui i raggi luminosi intersecano l'asse ottico nel grafico. Nella configurazione del tripletto di Cooke, le distanze individuate da q_2 Eq.(28) e f_2 Eq.(31) fanno riferimento al secondo vertice della lente C (Cap.2.3). Per trovare il punto di intersezione con l'asse ottico sarà dunque necessario calcolare la posizione del vertice della lente C.

$$z_C = z_A + d_A + d_b + d_C + d_d + d_E = 27.78 \text{ mm} \quad (47)$$

Dove $z_A = 5 \text{ mm}$ è la posizione della prima lente mentre d_A, d_C, d_E sono gli spessori delle lenti e d_d, d_e sono gli spessori degli spazi d'aria, come indicato in tabella a Cap.(2.3.1). Si ottiene così la seguente espressione per la posizione di intersezione dei raggi con l'asse ottico:

$$f_z = z_C + f_2 - h_2 = z_C + q_2 \quad (48)$$

Applicando Eq.(28), trovo che il valore di q_2 per il sistema considerato è: 85.52 mm. Si ottene in conclusione il seguente risultato:

$$f_z = 113.30 \text{ mm} \quad (49)$$

Come in precedenza, prima di confrontare questo risultato con la rappresentazione grafica si sottrae la posizione del secondo piano principale.

$$h_z = z_0 + dz + h_2 = 40.49 \text{ mm} \quad (50)$$

Ottenendo quindi la posizione del grafico f_z in cui i raggi luminosi vengono fatti convergere:

$$f_z = f_2 - h_z = 151.36 - 78.53 = 72.83 \text{ mm} \quad (51)$$

Come si può osservare il risultato è lo stesso di Eq.(46), risultato atteso viste le stesse posizioni dei

piani principali e del punto focale per i due sistemi. Nuovamente si osserva che i raggi vengono fatti convergere nella posizione del punto focale qui calcolata e già rappresentata in Fig.(20), provando ancora una volta l'accordo fra i risultati del modello matriciale e il ray tracing. Si è già visto come questi risultati risultino solo in prima approssimazione coerenti con quanto calcolato col metodo matriciale. È dunque possibile visualizzare nel grafico di ingrandimento attorno al punto di fuoco in Fig.(21) l'allontanamento dei risultati ottenuti dalla simulazione basata sul ray tracing rispetto ai risultati dell'ottica matriciale (Cap.2):

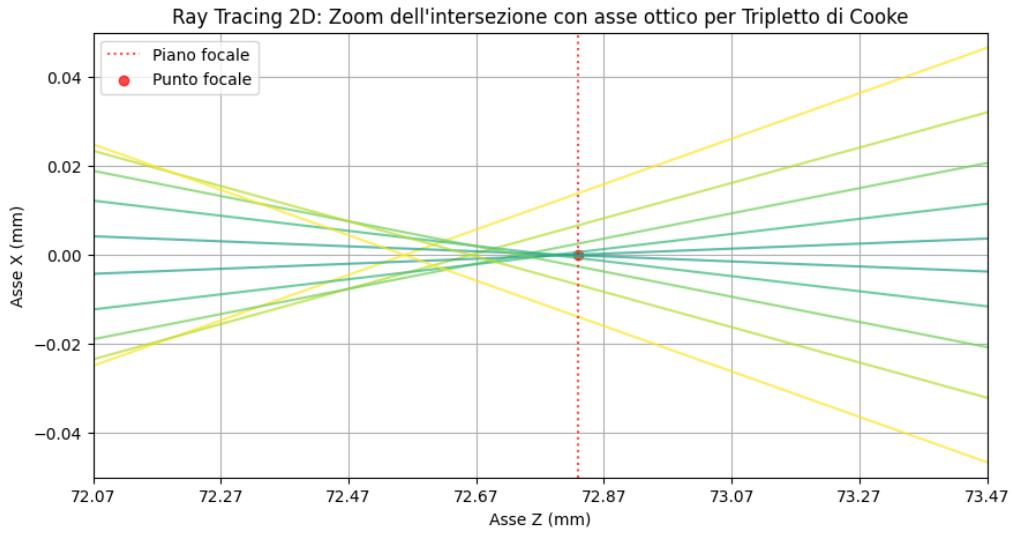


Figura 21: Ingrandimento in corrispondenza della distanza focale in funzione della distanza dal secondo piano principale.

Si ricorda che i valori di distanza focale ottenuti in Eq.(29) sono valide approssimazioni della posizione in cui i raggi intersecano l'asse ottico solamente nel caso di raggi parassiali. In particolare, i raggi che si discostano maggiormente dalla distanza focale teorica sono quelli che colpiscono la lente a un'altezza maggiore rispetto all'asse ottico, perciò i raggi gialli in Fig.(20). Al contrario, i raggi più scuri, il cui angolo di incidenza sulla lente è quasi trascurabile, rispettano quasi perfettamente il comportamento teorico atteso. Il risultato è coerente con quanto si ottiene nel caso nella lente spessa, un sistema più semplice che nonostante ciò viene caratterizzato dal fenomeno delle aberrazioni sferiche.

Questi allontanamenti rispetto al comportamento ideale danno vita al fenomeno delle aberrazioni ottiche. Per analizzare l'entità delle aberrazioni ottiche nei due casi studiamo la posizione in cui vengono fatti convergere un insieme arbitrario di raggi per i due sistemi considerati. I seguenti grafici, Fig.(22), Fig.(23) rappresentano dunque la distribuzione dei punti di intersezione fra un insieme arbitrario di raggi entranti ed il piano focale. Una volta generati casualmente nel piano normale all'asse ottico, ai raggi sono stati fatti percorrere i due diversi sistemi. Misurandone il percorso attraverso il ray tracing si è quindi calcolato il punto di incidenza del raggio nel piano focale. Ottenendo così le seguenti distribuzioni:

Distribuzione dei punti di intersezione dei raggi con l'asse focale per la lente spessa

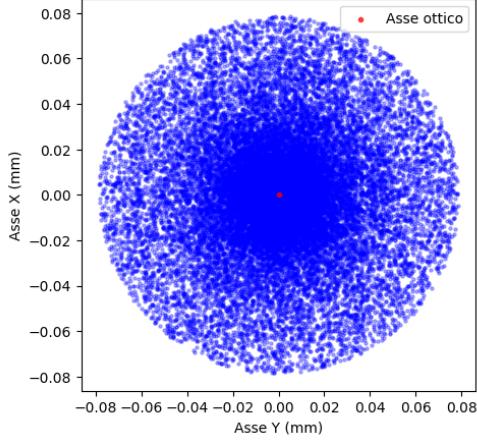


Figura 22: Distribuzione dei punti di intersezione dei raggi luminosi con l'asse focale nel caso della lente spessa.

Distribuzione dei punti di intersezione dei raggi con l'asse focale per il tripletto di Cooke

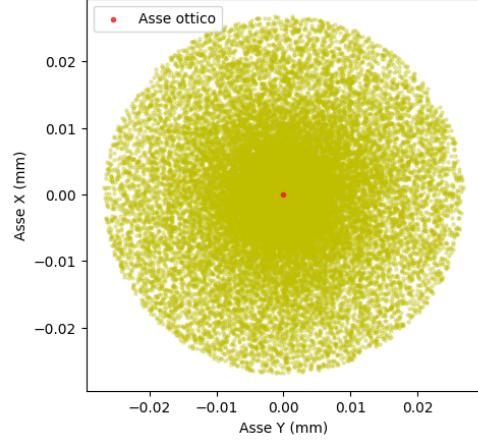


Figura 23: Distribuzione dei punti di intersezione dei raggi luminosi con l'asse focale nel caso del Tripletto di Cooke.

Nelle analisi si avrà che i grafici rappresentanti risultati per il tripletto di Cooke saranno gialli, mentre per la lente spessa saranno blu. Dai grafici ottenuti si osserva che la forma delle due distribuzioni è pressoché identica per i due sistemi. La differenza significativa che si riscontra sta nel range delle distribuzioni. Il seguente grafico permette di visualizzare un confronto diretto fra le distanze rispetto all'asse ottico a cui vengono fatti convergere i raggi per i due sistemi, permettendo di comprendere la luminosità che i sistemi ottici generano nel punto di convergenza. Si vede molto chiaramente in Fig.(24) che l'entità delle aberrazioni nel caso del tripletto di Cooke è notevolmente minore rispetto ad un sistema equivalente costituito da una lente spessa.

Distribuzione dei punti di intersezione dei raggi con l'asse focale per il tripletto di Cooke e la lente spessa

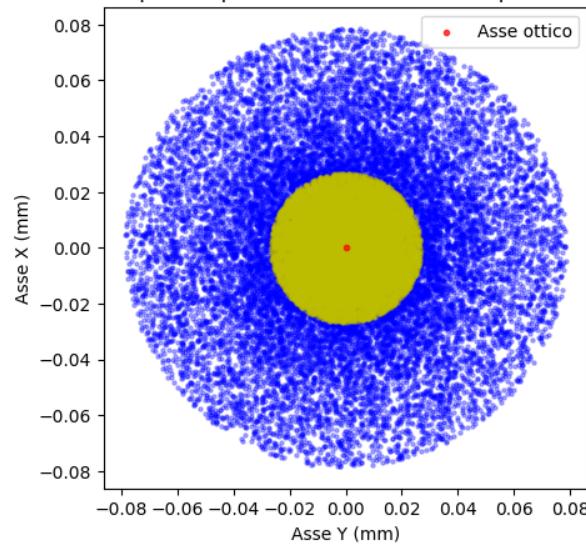


Figura 24: Confronto fra le distribuzioni dei punti di intersezione dei raggi luminosi con l'asse focale.

Il valore misurato di luminosità cala in funzione della distanza dall'asse ottico, in un sistema ideale la luminosità sarebbe nulla per posizioni diverse rispetto al punto focale. Per analizzare la luminosità dei sistemi ottici in funzione della distanza radiale rispetto all'asse ottico verrà studiata l'ampiezza a mezza altezza (FWHM) dei seguenti istogrammi.

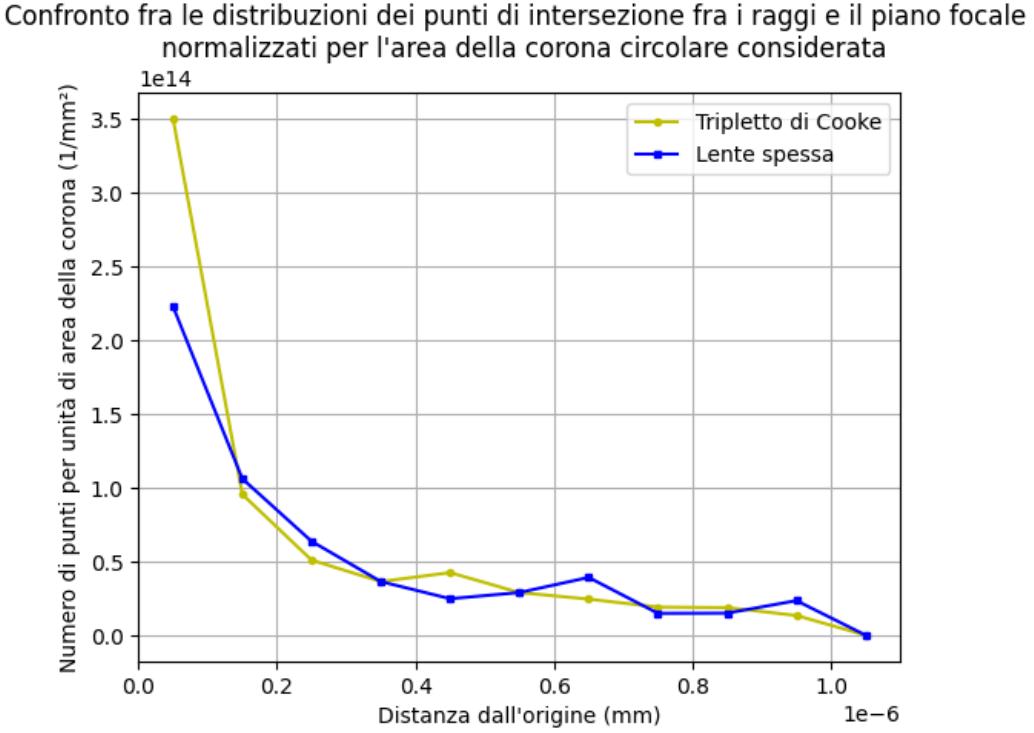


Figura 25: Confronto fra gli istogrammi delle dispersioni dei punti di focalizzazione.

Le imperfezioni dei sistemi sono facilmente evidenziabili dalla forma del grafici, le code sono rappresentative del valore delle aberrazioni. La FWHM fornisce informazioni sulla dispersione e sulla larghezza effettiva dei dati, mostrando quanto sono sparsi intorno al picco di massimo. La FWHM misura la larghezza della distribuzione a metà della sua altezza massima. In altre parole indica a quale distanza dal centro della distribuzione i dati si sono ridotti alla metà del valore massimo. Con queste informazioni è possibile comprendere quanto i dati siano distribuiti attorno al picco della distribuzione. Una FWHM piccola indica che i dati sono più concentrati vicino al valore centrale, mentre una FWHM più grande indica una maggiore dispersione. Quello che ci aspettiamo è un valore minore per l'altezza a mezza altezza dell'istogramma corrispondente ai valori del tripletto di Cooke, rispetto all'istogramma dei valori per la lente spessa Fig.(25). I valori di altezza a mezza altezza che si ottengono per il tripletto di Cooke e per la lente spessa sono in ordine:

$$\text{FWHM} = 0.026 \text{ mm} \quad (52)$$

$$\text{FWHM} = 0.078 \text{ mm} \quad (53)$$

Come aspettato la FWHM mostra che il sistema descritto dal tripletto di Cooke è più efficiente per la focalizzazione dei raggi incidenti. Il motivo per cui si attendevano valori minori per il tripletto di Cooke è la natura di questo sistema ottico, noto per l'elevata efficienza nella focalizzazione dei raggi e per le ottimizzazioni che possono essere effettuate per migliorare l'efficacia dell'obiettivo.

3.4 Aberrazioni cromatiche e ray tracing

Il ray tracing risulta particolarmente adatto alla trattazione delle aberrazioni cromatiche, permette infatti di analizzare correttamente il comportamento della luce per ciascuna lunghezza d'onda, tenendo conto delle variazioni dell'indice di rifrazione del materiale ottico (variazioni descritte generalmente dalla relazione di Cauchy o di Sellmeier), evidenziando quindi le piccole differenze di percorso dei casi analizzati. Nel caso delle aberrazioni cromatiche, il ray tracing evidenzia il fenomeno della dispersione, i raggi di lunghezze d'onda più corte, come il blu, subiscono una deviazione maggiore rispetto a quelli di lunghezze d'onda più lunghe, come il rosso. Di conseguenza, in un sistema ottico non corretto per le aberrazioni cromatiche è possibile osservare frange colorate attorno all'immagine che viene generata. Il modello verrà applicato, simulando la propagazione di raggi monocromatici di diverse lunghezze d'onda (ad esempio, rosso, verde, blu), per poi unire i risultati ed effettuare un confronto finale. Questa tecnica offrirà una rappresentazione accurata delle aberrazioni cromatiche in sistemi ottici complessi come il tripletto di Cooke o la lente spessa. Vediamo inoltre che il ray tracing può essere utilizzato sia per analizzare le aberrazioni cromatiche in configurazioni già esistenti, sia per progettare lenti ottiche migliorate. Nel caso del tripletto di Cooke, un obiettivo fotografico composto da tre lenti, il ray tracing permette di analizzare l'effetto combinato delle lenti su diverse lunghezze d'onda, evidenziando come l'introduzione di materiali con indici di rifrazione differenti possa aiutare a minimizzare l'aberrazione cromatica.

Analizziamo ora le aberrazioni cromatiche per la lente spessa applicando il programma python al sistema descritto al (Cap.2.3). In Fig.(26) è possibile intravedere come il percorso dei raggi, diretti nel punto focale, sia differente per le varie lunghezze d'onda.

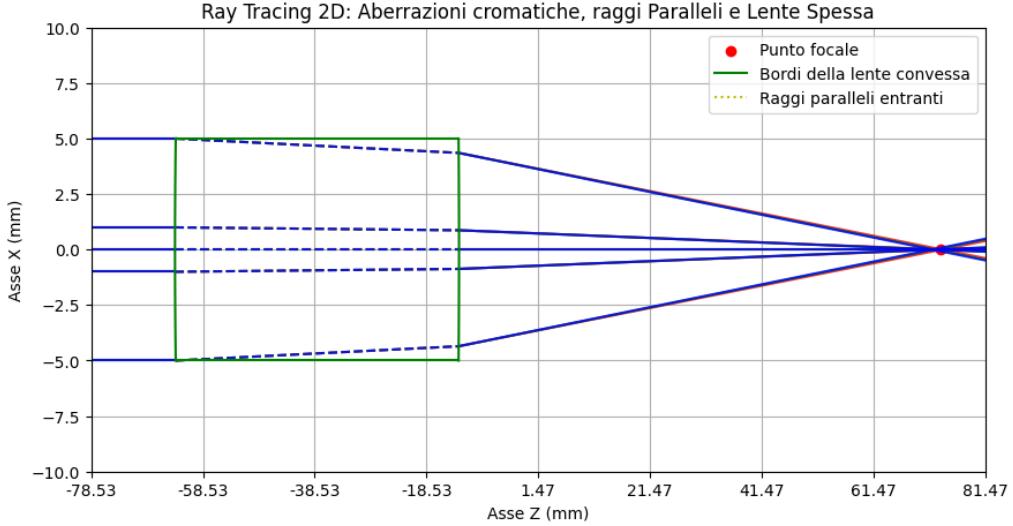


Figura 26: Ray tracing della lente spessa per diverse lunghezze d'onda della luce.

Si calcola quindi il punto focale a cui le diverse lunghezze d'onda della luce intersecano l'asse ottico. Identicamente a quanto fatto nei capitoli precedenti l'origine dell'asse Z è stata posta in corrispondenza della posizione del secondo piano principale dei sistemi ottici. Similmente a quanto fatto a (Cap.3.2) applicando Eq.(50) ed i risultati di Eq.(37) viene ricavata la posizione del vertice della lente, dei piani principali e quindi la posizione del punto focale. Ottenendo per le tre configurazioni il rispettivo risultato:

$$f_{z,\text{rosso}} = 74.04 \text{ mm} \quad f_{z,\text{verde}} = 72.82 \text{ mm} \quad f_{z,\text{blu}} = 72.23 \text{ mm} \quad (54)$$

Si verifica l'accordo con quanto atteso, in particolare si osserva una maggiore rifrazione per i raggi blu ed una rifrazione minore per i raggi rossi. È dunque possibile visualizzare nel grafico di ingrandimento attorno al punto di fuoco in Fig.(27) l'allontanamento dei risultati ottenuti dalla simulazione basata sul ray tracing rispetto ai risultati dell'ottica matriciale (Cap.2):

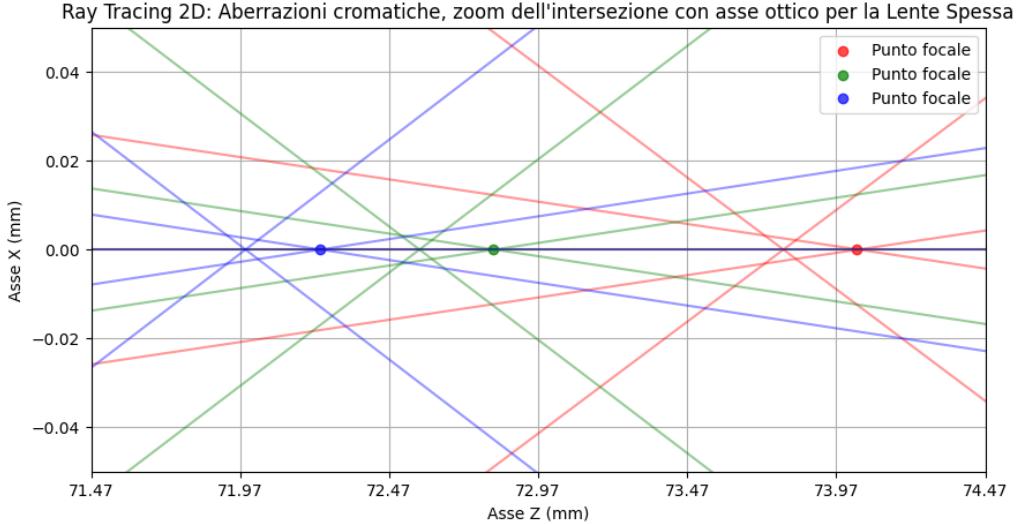


Figura 27: Ingrandimento in corrispondenza della distanza focale.

Abbiamo quindi dimostrato, sfruttando il ray tracing, la presenza delle aberrazioni cromatiche per la lente spessa. Effettuiamo quindi un’analisi quantitativa di queste aberrazioni ottenendo informazioni univoche da poter confrontare col caso del tripletto di Cooke, per il quale si ripetono le analisi appena proposte. Si osserva in Fig.(28) che il percorso dei raggi, diretti nel punto focale risultà in prima analisi più compatto rispetto al caso della lente spessa.

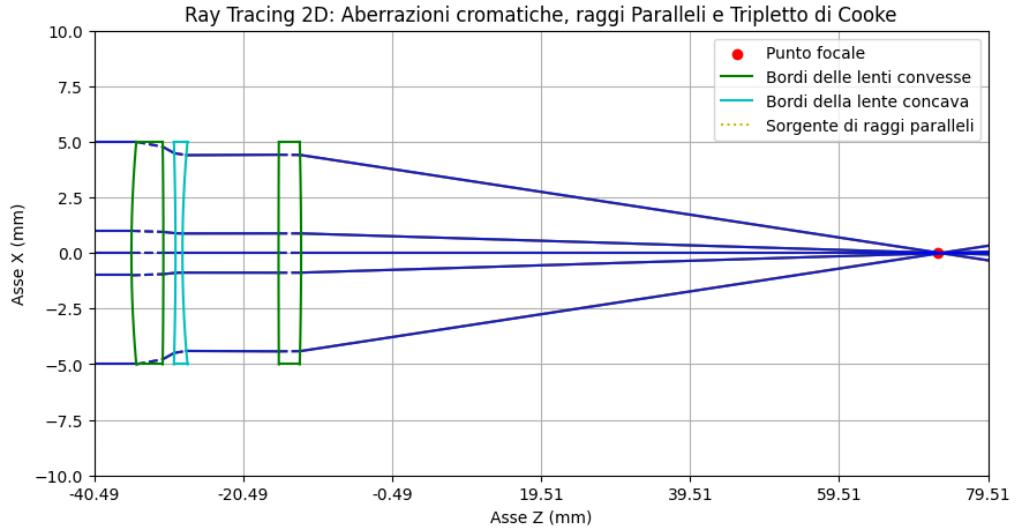


Figura 28: Ray tracing del tripletto di Cooke per diverse lunghezze d’onda della luce.

Per le tre configurazioni considerate, il punto focale a diverse lunghezze d’onda viene ottenuto applicando i risultati in Eq.(39) similmente a quanto già effettuato in Eq(48):

$$f_{z,\text{rosso}} = 72.68 \text{ mm} \quad f_{z,\text{verde}} = 72.70 \text{ mm} \quad f_{z,\text{blu}} = 72.90 \text{ mm} \quad (55)$$

Questi dati, fondamentali per generare correttamente l'immagine in Fig.(29) evidenziano sia il fenomeno dell'aberrazione sferica che cromatica.

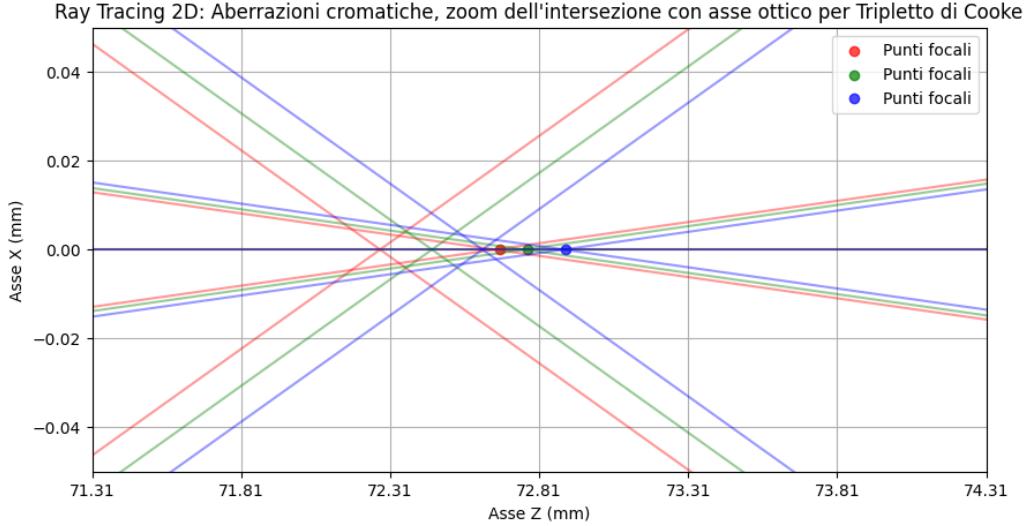


Figura 29: Ingrandimento in corrispondenza della distanza focale.

Notiamo un fenomeno interessante, infatti in questo caso i raggi maggiormente rifratti corrispondono alle lunghezze d'onda della luce rossa. Questo risultato è conseguenza della lente in vetro Flint che produce un effetto opposto alle lenti in vetro Crown, l'effetto opposto influisce maggiormente sul percorso dei raggi determinando una maggiore rifrazione per i raggi di lunghezza d'onda maggiore. Questa proprietà viene sfruttata nelle lenti apocromatiche per ridurre le aberrazioni cromatiche. Infatti ottimizzando le variabili del sistema è possibile, in linea teorica, generare effetti rifrattivi uguali e opposti per i raggi luminosi aventi diverse lunghezze d'onda, generando quindi delle immagini senza aberrazioni cromatiche.

Procediamo dunque confrontando i risultati ottenuti per i due sistemi ottici. I grafici in Fig.(30), Fig.(31) rappresentano la distribuzione dei punti di intersezione col piano focale per un insieme arbitrario di raggi entranti, aventi le tre lunghezze d'onda della luce precedentemente introdotte.

Aberrazioni cromatiche: Distribuzione dei punti di intersezione dei raggi con l'asse focale per la lente spessa

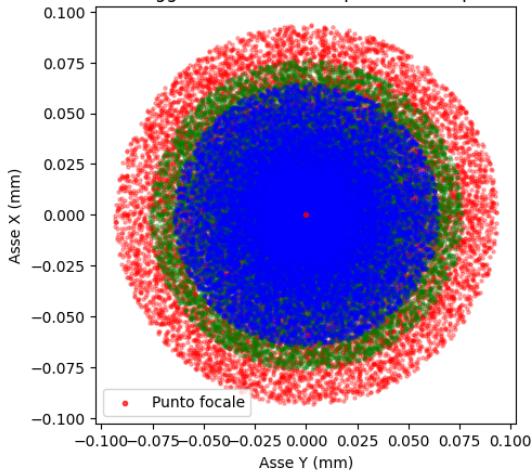


Figura 30: Distribuzione dei punti di intersezione dei raggi luminosi rosso, verde, blu, con l'asse focale per la lente spessa.

Aberrazioni cromatiche: Distribuzione dei punti di intersezione dei raggi con l'asse focale per il tripletto di Cooke

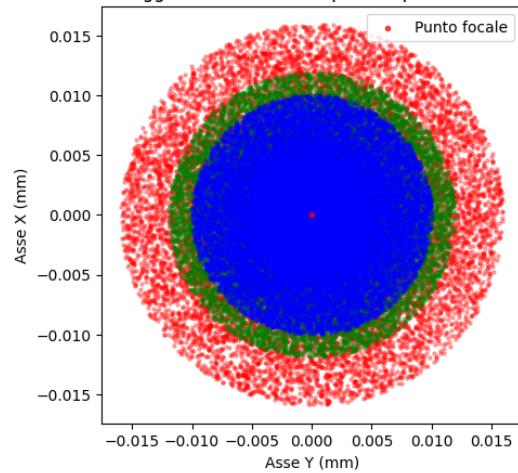


Figura 31: Distribuzione dei punti di intersezione dei raggi luminosi rosso, verde, blu, con l'asse focale per il Tripletto di Cooke.

Aberrazioni cromatiche: Confronto fra le distribuzione dei punti di intersezione dei raggi con l'asse focale per il tripletto di Cooke e la lente spessa

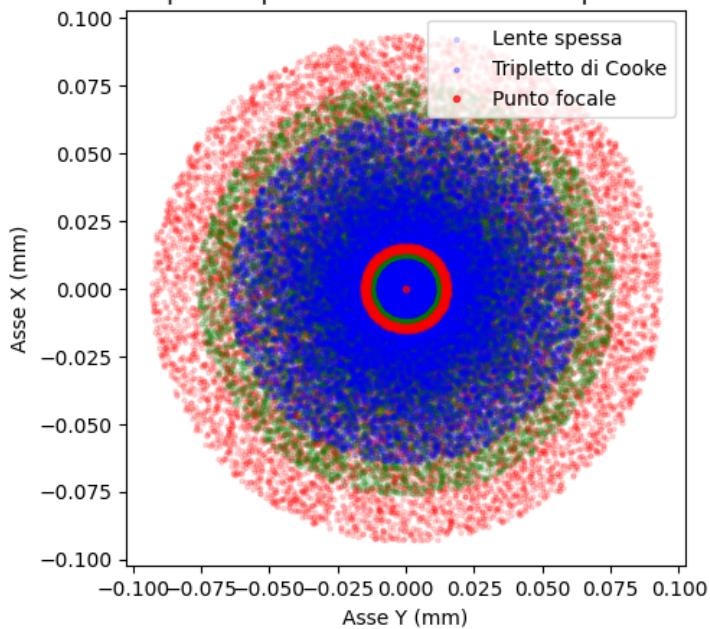


Figura 32: Confronto fra le distribuzioni dei punti di intersezione dei raggi luminosi rosso, verde, blu, con l'asse focale per i due sistemi.

Come si può vedere i raggi di colore blu risultano in immagini più luminose mentre le lunghezze d'onda di colore rosso determinano indici di rifrazione che portano ad una dispersione maggiore. La forma delle due distribuzioni è molto simile, per comprendere quale dei due sistemi non ottimizzati fornisce immagini in cui è presente minorellamente l'effetto delle aberrazioni cromatiche

la trattazione è necessario studiare l'ampiezza a mezza altezza delle distribuzioni. Si vede infatti che un confronto diretto fra i grafici, Fig.(32) evidenzia una significativa differenza nella larghezza delle aberrazioni e della zona illuminata. Procediamo quindi a studiare l'ampiezza a mezza altezza di queste configurazioni:

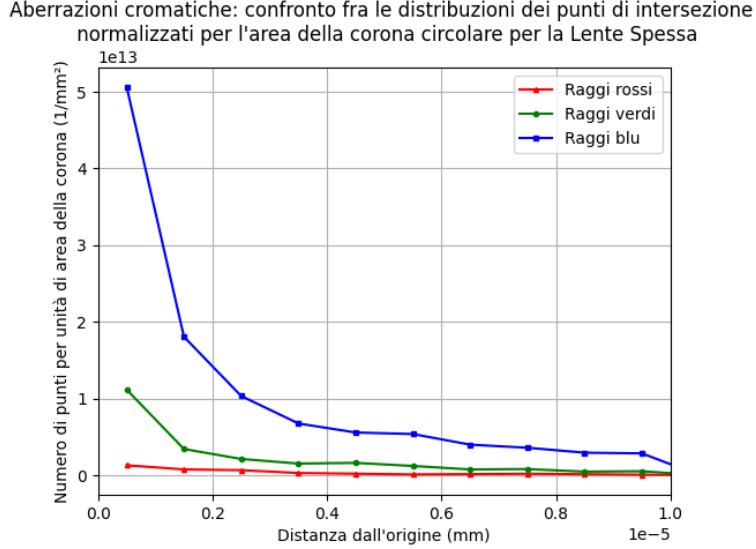


Figura 33: Istogramma delle dispersioni dei punti di focalizzazione in funzione della distanza dall'asse ottico per la Lente spessa e raggi luminosi di colore rosso, verde, blu.

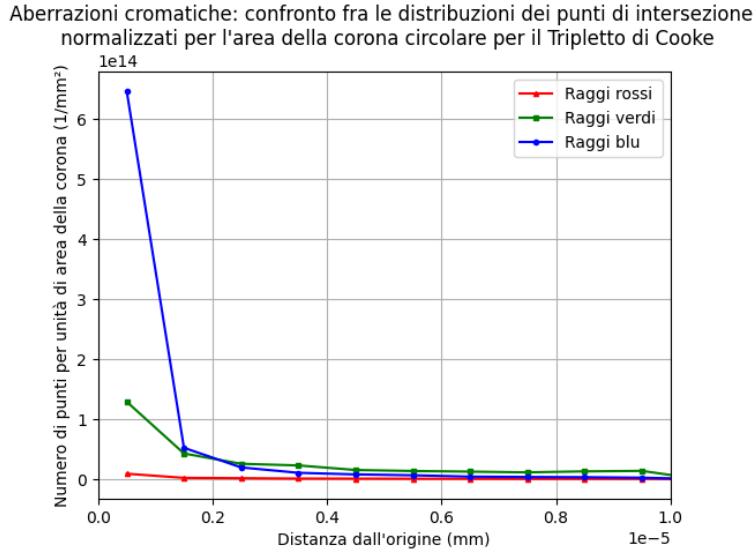


Figura 34: Istogramma delle dispersioni dei punti di focalizzazione in funzione della distanza dall'asse ottico per il Tripletto di Cooke e raggi luminosi di colore rosso, verde, blu.

Per confrontare i risultati valutiamo l'entità della differenza fra le FWHM per la luce di colore rosso e blu, si considerano questi colori per la luce in quanto forniscono una stima dell'ampiezza della dispersione per lunghezze d'onda agli estremi dello spettro. Sistemi ottici composti quali il

tripletto di Cooke composti da lenti con differente indice di rifrazione risultano predisposti ad essere ottimizzati creando sistemi apocromatici⁷. Si attende dunque un valore minore per la differenza l'altezza a mezza altezza dell'istogramma corrispondente ai valori del tripletto di Cooke Fig.(33), rispetto all'istogramma dei valori per la lente spessa Fig.(34). I valori di altezza a mezza altezza che si ottengono per le tre lunghezze d'onda della luce nel caso della lente spessa sono i seguenti:

$$\text{FWHM}_{ls,\text{rosso}} = 0.093 \text{ mm} \quad \text{FWHM}_{ls,\text{verde}} = 0.076 \text{ mm} \quad \text{FWHM}_{ls,\text{blu}} = 0.066 \text{ mm} \quad (56)$$

Per il tripletto di Cooke si ottiene invece:

$$\text{FWHM}_{tc,\text{rosso}} = 0.015 \text{ mm} \quad \text{FWHM}_{tc,\text{verde}} = 0.011 \text{ mm} \quad \text{FWHM}_{tc,\text{blu}} = 0.009 \text{ mm} \quad (57)$$

Si vede in definitiva che la differenza di FWHM per i due sistemi considerati vale:

$$\Delta\text{FWHM}_{ls} = 0.027 \text{ mm} \quad (58)$$

$$\Delta\text{FWHM}_{tc} = 0.006 \text{ mm} \quad (59)$$

Risulta evidente come il sistema ottico del tripletto di Cooke fornisca risultati caratterizzati da un'aberrazione cromatica minore rispetto al caso della lente spessa. La differenza fra le larghezze a mezza altezza determina risultati che evidenziano in modo equivoco la distinzione fra i due sistemi e la superiorità del tripletto di Cooke non ottimizzato nel generare risultati caratterizzati da una minore dispersione, verificando come anche per le aberrazioni cromatiche il sistema composto costituisce un sistema più efficace per focalizzare la luce.

⁷I sistemi apocromatici sono configurazioni ottiche che correggono l'aberrazione cromatica bilanciando la dispersione del materiale ottico attraverso l'uso di lenti con diverse proprietà rifrattive

4 Conclusioni

In conclusione, per due sistemi ottici equivalenti, lente spessa e tripletto di Cooke, dotati della stessa distanza focale e posizione dei piani principali, è stata evidenziata una significativa differenza nell'efficacia con cui i raggi ottici vengono focalizzati. I grafici ottenuti in Fig.(24) hanno confermato i risultati attesi, in particolare è stato evidenziato come per una lente spessa ed un tripletto di Cooke aventi gli stessi valori di distanza focale e posizione dei piani principali, il tripletto di Cooke focalizza i raggi entranti con un'accuratezza significativamente migliore. Questo risultato conferma l'efficacia del tripletto di Cooke e giustifica la notorietà di questo sistema composto che dopo più di cent'anni dalla sua ideazione trova ancora moltissime applicazioni. Lo studio dell'ampiezza a mezza altezza ha infine fornito dei riscontri numerici per comprendere le differenze nelle distribuzioni nei due casi. In Eq.(52) ed Eq.(53) è stato evidenziato come i valori di FWHM per la lente spessa siano nettamente maggiori dei corrispettivi valori per il tripletto di Cooke, che quindi rappresenta un sistema di qualità migliore perché capace di focalizzare i raggi con una dispersione minore. Il grafico in Fig.(25) paragona direttamente i risultati ottenuti per i due casi considerati, in blu sono riportati i dati per la lente spessa e in giallo i dati per il tripletto di Cooke. Da questo confronto diretto si osserva visivamente il risultato descritto dai valori delle ampiezze a mezza altezza. Infatti si può vedere come la dispersione dei valori nel caso della lente spessa sia un fenomeno molto più presente rispetto al caso del tripletto. Inoltre si può osservare, come in vicinanza del punto focale, la luminosità del tripletto di Cooke sia nettamente maggiore.

Si è inoltre visto che l'analisi comparativa tra la lente spessa e il tripletto di Cooke ha evidenziato differenze significative nella qualità dell'immagine anche per il caso delle aberrazioni ottiche cromatiche. Utilizzando il metodo della FWHM per analizzare le distribuzioni dei raggi luminosi focalizzati sui piani di immagine, è stato possibile quantificare in modo preciso l'entità delle aberrazioni nei due sistemi ottici. I risultati in Eq.(58), Eq.(59) mostrano chiaramente che la maggiore larghezza della distribuzione dei raggi luminosi per la lente spessa, misurata attraverso la FWHM, corrisponde ad aberrazioni cromatiche e sferiche più marcate, quindi una dispersione più ampia dei raggi e una minore capacità di concentrare la luce in un singolo punto focale. Questo fenomeno è associato sia a un incremento dell'aberrazione cromatica longitudinale, dovuto alla variazione dell'indice di rifrazione con la lunghezza d'onda, sia a una maggiore aberrazione sferica, causata dalla geometria più semplice della lente spessa. Al contrario, il tripletto di Cooke, grazie alla sua progettazione più complessa, che utilizza materiali con indici di rifrazione diversi e una disposizione più sofisticata delle lenti, riesce a correggere in modo più efficace le aberrazioni. I valori di FWHM ottenuti per il tripletto sono significativamente più piccoli, dimostrando una migliore capacità di focalizzazione e una riduzione delle frange cromatiche attorno all'immagine. Il risultato risulta particolarmente significativo se confrontato con i più comuni sistemi di lenti apocromatiche. Si vede in figura Fig.(35), come questi sistemi complessi possono essere ottenuti dall'ottimizzazione di sistemi ottici quali il tripletto di Cooke. Infatti, indici di diffrazioni diversi per alcune delle lenti dentro al sistema permettono di usare lenti che si comportano in modo diverso per la stessa lunghezza d'onda.

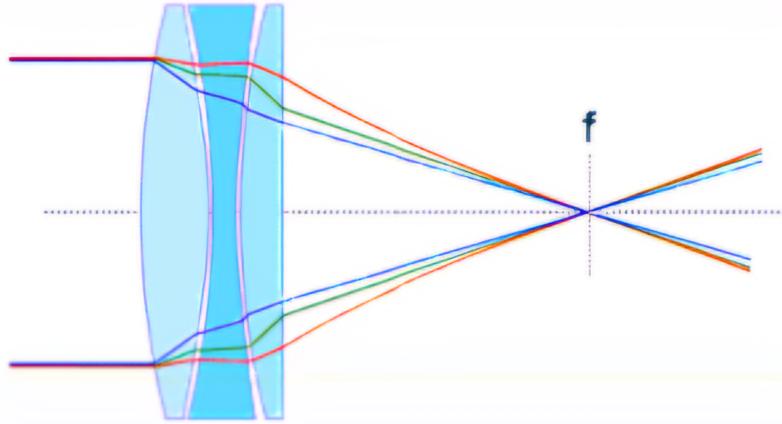


Figura 35: Esempio di sistema ottico apocromatico ottenuto dal tripletto di Cooke.

Questi risultati confermano l’importanza di una progettazione ottica avanzata, come quella del tripletto di Cooke, capace di ridurre le aberrazioni e migliorare la qualità dell’immagine. L’approccio metodologico utilizzato, basato sia sul ray tracing che sull’analisi matriciale per raggi parassiali, si è rivelato efficace nel descrivere e confrontare il comportamento dei due sistemi ottici in esame, fornendo informazioni dettagliate sull’entità delle aberrazioni sferiche e cromatiche e sulla loro influenza sulle prestazioni globali del sistema.

Il lavoro ha quindi posto le basi al progetto di ottimizzazione che, a partire dalla rappresentazione matriciale già ottenuta e dai programmi di ray tracing, permetterà di eliminare le aberrazioni sferiche. Un diretto esempio dell’applicazione dei risultati ottenuti nell’ottimizzazione di un tripletto di Cooke viene riportato in [10]. Il lavoro svolto ha fornito risultati soddisfacenti nella descrizione di sistemi composti e ha definito dei modelli facilmente applicabili per l’individuazione e misura delle aberrazioni sferiche dei sistemi ottici. Ed è stato inoltre in grado di fornire un diretto confronto fra l’efficienza nel focalizzare i raggi entranti per due sistemi ottici equivalenti quali una specia lente spessa ed il tripletto di Cooke.

5 Bibliografia

Riferimenti bibliografici

- [1] Hecht, E. (2017). Optics. (5th ed.) Pearson, .
- [2] Guenther, B. D. (2004). Modern Optics Simplified. SPIE Press.
- [3] Hecht, E. (2002). Optics (4th ed.). Addison-Wesley.
- [4] Pedrotti, F. L., and Pedrotti, L. S. (2017). Introduction to Optics. (3rd ed.) Pearson.
- [5] Sundnes, J. (2010). Introduction to Scientific Programming with Python. Springer Open.
- [6] Goodman, Douglas S. General Principles of Geometric Optics.
- [7] Morgan Kaufmann. (2019). An Introduction to Ray Tracing (Version 1.3).
- [8] Saleh, B. E. A., and Teich, M. C. (1991). Fundamentals of Photonics. John Wiley and Sons, Inc.
- [9] University of California, Ernest O. Lawrence Radiation Laboratory. (1987). Report (Ricevuto il 21 agosto 1987). Lawrence Berkeley National Laboratory.
- [10] Phillip K Poon. (2017). Designing a Cooke Triplet. <http://phillipkpoon.net/jekyll/update/2017/02/27/Designing-A-Cooke-Triplet.html>
- [11] Maiello, A. (2003, ottobre). Tutti noi diamo per scontato che dentro gli obiettivi ci siano lenti di vetro. Perché? Nadir. urly.it/310yn1.
- [12] Wolfram Alpha. (2023). *Wolfram Alpha: Computational Intelligence*. Retrieved from <https://www.wolframalpha.com>.
- [13] Refractive Index Database. "N-SK2 - SCHOTT". Available: <https://refractiveindex.info/?shelf=glass&book=SCHOTT-SK&page=N-SK2>.

A Appendice

A.1 Codice Python per il ray tracing del tripletto di Cooke

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # definizione dei funzionali:
5
6 # propagazione attraverso un mezzo
7 def calcolo_propagazione(altezza, angolo, distanza, indice_rifrazione):
8     return altezza + distanza / indice_rifrazione *np.tan(angolo)
9
10 # rifrazione su una superficie
11 def calcolo_rifrazione(altezza, angolo, indice_rifrazione_iniziale,
12     indice_rifrazione_finale, raggio_curvatura):
13     return np.arcsin(indice_rifrazione_iniziale/indice_rifrazione_finale
14         * np.sin(angolo + np.arcsin(altezza/raggio_curvatura))) - np.
15         arcsin(altezza/raggio_curvatura)
16
17 # intercetta della retta
18 def intercetta(x0, y0, m):
19     return y0-x0*m
20
21 # Definizione della retta
22 m = 1 # coefficiente angolare
23 q = 0 # intercetta
24 def retta(x):
25     return m * x + q
26
27 # Trova i punti di intersezione tra retta e circonferenza
28 def punti_intersezione(m, q, h, k, r): # h: centro coordinate x, k:
29     centro coordinate y della circonferenza
30     # Equazione della circonferenza:  $(x-h)^2 + (y-k)^2 = r^2$ 
31     # Sostituisco y con l'equazione della retta  $y = mx + q$ :
32     #  $(x-h)^2 + (mx + q - k)^2 = r^2$ 
33     A = 1 + m**2
34     B = 2*m*(q - k) - 2*h
35     C = h**2 + (q - k)**2 - r**2
36
37     discriminante = B**2 - 4*A*C
38
39     if discriminante < 0:
40         return None # Nessuna intersezione
41     elif discriminante == 0:
42         x = -B / (2*A)
43         y = retta(x)
44         return [(x, y)] # Un solo punto di intersezione
45     else:
46         x1 = (-B + np.sqrt(discriminante)) / (2*A)
```

```

43         y1 = retta(x1)
44         x2 = (-B - np.sqrt(discriminante)) / (2*A)
45         y2 = retta(x2)
46         return [(x1, y1), (x2, y2)] # Due punti di intersezione
47
48
49 # Parametri iniziali
50 x = 0
51 y = 0
52 intersezioni_x = []
53
54
55 #indici di rifrazione
56 n0 = 1.000 # Indice di rifrazione dell'aria
57 n1 = 1.6110 # Indice di rifrazione della lente crown
58 n2 = 1.5744 # Indice di rifrazione della lente flint
59
60 #lente A
61 ya = 10 # Altezza della lente
62 xa = 5 # Posizione x della lente
63 dax = 4.29 # Spessore della lente
64 ha = 0 # posizione y della lente
65 ra1 = 19.4 #raggio di curvatura prima superficie
66 ra2 = -128.3 #raggio di curvatura seconda superficie
67
68
69 #lente B
70 yb = 10 # Altezza della lente
71 xb = xa+dax+1.63 # Posizione x della lente
72 dbx = 0.93 # Spessore della lente
73 hb = 0 # posizione y della lente
74 rb1 = -57.8 #raggio di curvatura prima superficie
75 rb2 = 18.9 #raggio di curvatura seconda superficie
76
77
78 #lente C
79 yc = 10 # Altezza della lente
80 xc = xb+dbx+12.9 # Posizione x della lente
81 dcx = 3.03 # Spessore della lente
82 hc = 0 # posizione y della lente
83 rc1 = 311.3 #raggio di curvatura prima superficie
84 rc2 = -66.4 #raggio di curvatura seconda superficie
85
86
87 # inizializzazione dei vettori utilizzati nel calcolo del percorso
88 # vengono definiti tanti diversi vettori per rendere pi intutitivo il
     calcolo
89
90 raggio_a1= np.linspace(0, 0, 2)
91 raggio_a2= np.linspace(0, 0, 2)

```

```

92 raggio_av= np.linspace(0, 0, 2)
93 raggio_b1= np.linspace(0, 0, 2)
94 raggio_b2= np.linspace(0, 0, 2)
95 raggio_bv= np.linspace(0, 0, 2)
96 raggio_c1= np.linspace(0, 0, 2)
97 raggio_c2= np.linspace(0, 0, 2)
98 raggio_cv= np.linspace(0, 0, 2)
99 raggio_d= np.linspace(0, 0, 2)
100 raggio_e= np.linspace(0, 0, 2)
101 raggio_d= np.linspace(0, 0, 2)
102 raggio_e= np.linspace(0, 0, 2)
103 raggio_f= np.linspace(0, 0, 2)
104 raggio_g= np.linspace(0, 0, 2)

105
106
107 n_raggi = 10 # Numero di raggi paralleli
108 altezza_massima = ya # Altezza massima a cui entrano i raggi
109
110 # Genera n_raggi paralleli orizzontali con diverse altezze
111 altezze = np.linspace(-altezza_massima/2, altezza_massima/2, n_raggi)
112 raggi_iniziali = np.array([[h, 0] for h in altezze])

113
114 # Inizia il grafico (una sola figura)
115 plt.figure(figsize=(10, 5))

116
117 # colori dei raggi
118 i=10
119 k=10
120 colormap = plt.cm.get_cmap('viridis', len(raggi_iniziali))

121
122 for raggio in raggi_iniziali:
123     # colori dei raggi
124     if(5<k<=10):
125         i=i-1
126         k=k-1
127     elif(k==5):
128         i=5
129         k=k-1
130     else:
131         i=i+1
132         k=k-1

133
134     # sono interessato a trovare i punti di intersezione per la retta:
135     qa1=intercetta(0, raggio[0], np.tan(raggio[1])/n0)
136     intersezioni_a1 = punti_intersezione(np.tan(raggio[1])/n0, qa1, xa+
137                                         ra1, ha, ra1)

138     # calcolo del raggio dopo lo spazio libero D
139     raggio_d[0] = calcolo_propagazione(raggio[0], raggio[1],
140                                         intersezioni_a1[1][0], n0)

```

```

140 raggio_d[1] = raggio[1]
141
142 # Calcolo del raggio dopo la prima lente A
143 raggio_a1[0] = raggio_d[0]
144 raggio_a1[1] = calcolo_rifrazione(raggio_d[0], raggio_d[1], n0, n1,
145     ra1)
146
147 qa2=intercetta(intersezioni_a1[1][0], raggio_a1[0], np.tan(raggio_a1
148     [1])/n1)
149 intersezioni_a2 = punti_intersezione(np.tan(raggio_a1[1])/n1, qa2,
150     xa+dax+ra2, ha, ra2)
151
152 # Calcolo del raggio dopo lo spazio nel vetro A
153 raggio_av[0] = calcolo_propagazione(raggio_a1[0], raggio_a1[1],
154     intersezioni_a2[0][0]-intersezioni_a1[1][0], n1)
155 raggio_av[1] = raggio_a1[1]
156
157 # Calcolo del raggio dopo la seconda lente A
158 raggio_a2[0] = raggio_av[0]
159 raggio_a2[1] = calcolo_rifrazione(raggio_av[0], raggio_av[1], n1, n0
160     , ra2)
161
162 qb1=intercetta(intersezioni_a2[0][0], raggio_a2[0], np.tan(raggio_a2
163     [1])/n0)
164 intersezioni_b1 = punti_intersezione(np.tan(raggio_a2[1])/n0, qb1,
165     xb+rb1, hb, rb1)
166
167 # calcolo della propagazione nel vuoto dopo la seconda lente E
168 raggio_e[0] = calcolo_propagazione(raggio_a2[0], raggio_a2[1],
169     intersezioni_b1[0][0]-intersezioni_a2[0][0] , n0)
170 raggio_e[1] = raggio_a2[1]
171
172 # Calcolo del raggio dopo la prima lente B
173 raggio_b1[0] = raggio_e[0]
174 raggio_b1[1] = calcolo_rifrazione(raggio_e[0], raggio_e[1], n0, n2,
175     rb1)
176
177 qb2=intercetta(intersezioni_b1[0][0], raggio_b1[0], np.tan(raggio_b1
178     [1])/n2)
179 intersezioni_b2 = punti_intersezione(np.tan(raggio_b1[1])/n2, qb2,
180     xb+dbx+rb2, hb, rb2)
181
182 # Calcolo del raggio dopo lo spazio nel vetro B
183 raggio_bv[0] = calcolo_propagazione(raggio_b1[0], raggio_b1[1],
184     intersezioni_b2[1][0]-intersezioni_b1[0][0] , n2)
185 raggio_bv[1] = raggio_b1[1]
186
187 # Calcolo del raggio dopo la seconda lente B
188 raggio_b2[0] = raggio_bv[0]

```

```

177 raggio_b2[1] = calcolo_rifrazione(raggio_bv[0], raggio_bv[1], n2, n0
178     , rb2)
179 qc1=intercetta(intersezioni_b2[1][0], raggio_b2[0], np.tan(raggio_b2
180     [1])/n0)
180 intersezioni_c1 = punti_intersezione(np.tan(raggio_b2[1])/n0, qc1,
181     xc+rc1, hc, rc1)
182
182 # calcolo della propagazione nel vuoto dopo la seconda lente F
183 raggio_f[0] = calcolo_propagazione(raggio_b2[0], raggio_b2[1],
184     intersezioni_c1[1][0]-intersezioni_b2[1][0] , n0)
184 raggio_f[1] = raggio_b2[1]
185
186 # Calcolo del raggio dopo la prima lente C
187 raggio_c1[0] = raggio_f[0]
188 raggio_c1[1] = calcolo_rifrazione(raggio_f[0], raggio_f[1], n0, n1,
189     rc1)
190
190 qc2=intercetta(intersezioni_c1[1][0], raggio_c1[0], np.tan(raggio_c1
191     [1])/n1)
191 intersezioni_c2 = punti_intersezione(np.tan(raggio_c1[1])/n1, qc2,
192     xc+dcx+rc2, hc, rc2)
193
193 # Calcolo del raggio dopo lo spazio nel vetro C
194 raggio_cv[0] = calcolo_propagazione(raggio_c1[0], raggio_c1[1],
195     intersezioni_c2[0][0]-intersezioni_c1[1][0] , n1)
195 raggio_cv[1] = raggio_c1[1]
196
197 # Calcolo del raggio dopo la seconda lente C
198 raggio_c2[0] = raggio_cv[0]
199 raggio_c2[1] = calcolo_rifrazione(raggio_cv[0], raggio_cv[1], n1, n0
200     , rc2)
201
201 # calcolo della propagazione nel vuoto dopo la seconda lente G
202 raggio_g[0] = calcolo_propagazione(raggio_c2[0], raggio_c2[1], 100+2
203     , n0)
203 raggio_g[1] = raggio_c2[1]
204
205 # Visualizzazione del percorso del raggio
206 plt.plot([0, intersezioni_a1[1][0]], [raggio[0], raggio_d[0]], color
207     =colormap(i), alpha=0.8)
207 plt.plot([intersezioni_a1[1][0], intersezioni_a2[0][0]], [raggio_d
208     [0], raggio_av[0]], linestyle='--', color=colormap(i), alpha=0.8)
208 plt.plot([intersezioni_a2[0][0], intersezioni_b1[0][0]], [raggio_av
209     [0], raggio_e[0]], color=colormap(i), alpha=0.8)
209 plt.plot([intersezioni_b1[0][0], intersezioni_b2[1][0]], [raggio_e
210     [0], raggio_bv[0]], linestyle='--', color=colormap(i), alpha=0.8)
210 plt.plot([intersezioni_b2[1][0], intersezioni_c1[1][0]], [raggio_bv
210     [0], raggio_f[0]], color=colormap(i), alpha=0.8)

```

```

211     plt.plot([intersezioni_c1[1][0], intersezioni_c2[0][0]], [raggio_f
212             [0], raggio_cv[0]], linestyle='--', color=colormap(i), alpha=0.8)
213     plt.plot([intersezioni_c2[0][0], intersezioni_c2[0][0] + 100 + 2], [
214             raggio_cv[0], raggio_g[0]], color=colormap(i), alpha=0.8)
215
216     # grafico le intersezioni con y=0
217     m_c = raggio_g[1] # Pendenza dopo la lente C
218     y_c = raggio_c2[0] # Altezza dopo la lente C
219     x_intersezione_c = (0 - y_c) / m_c + intersezioni_c2[0][0]
220     print(f"inclinazione raggio {i},{m_c}, altezza iniziale:{y_c},
221           posizione risultante: {x_intersezione_c}")
222     intersezioni_x.append(x_intersezione_c)
223
224
225 # Disegna i bordi della lente A
226 theta1a = np.linspace(-np.arcsin((ya/2)/ra1), np.arcsin((ya/2)/ra1),
227                       100)
228 theta2a = np.linspace(-np.arcsin((ya/2)/ra2), np.arcsin((ya/2)/ra2),
229                       100)
230
231 x_lente1a = xa + ra1 * (1 - np.cos(theta1a))
232 y_lente1a = ha + ra1 * np.sin(theta1a)
233
234 x_lente2a = xa + dax + ra2 * (1 - np.cos(theta2a))
235 y_lente2a = ha + ra2 * np.sin(theta2a)
236
237 plt.plot(x_lente1a, y_lente1a, color='g')
238 plt.plot(x_lente2a, y_lente2a, color='g')
239
240 # Disegna i bordi della lente B
241 theta1b = np.linspace(-np.arcsin((yb/2)/rb1), np.arcsin((yb/2)/rb1),
242                       100)
243 theta2b = np.linspace(-np.arcsin((yb/2)/rb2), np.arcsin((yb/2)/rb2),
244                       100)
245
246 x_lente1b = xb + rb1 * (1 - np.cos(theta1b))
247 y_lente1b = hb + rb1 * np.sin(theta1b)
248
249 x_lente2b = xb + dbx + rb2 * (1 - np.cos(theta2b))
250 y_lente2b = hb + rb2 * np.sin(theta2b)
251
252 plt.plot(x_lente1b, y_lente1b, color='c')
253 plt.plot(x_lente2b, y_lente2b, color='c')
254
255 # Disegna i bordi della lente C
256 theta1c = np.linspace(-np.arcsin((yc/2)/rc1), np.arcsin((yc/2)/rc1),
257                       100)
258 theta2c = np.linspace(-np.arcsin((yc/2)/rc2), np.arcsin((yc/2)/rc2),
259                       100)

```

```

252 x_lente1c = xc + rc1 * (1 - np.cos(theta1c))
253 y_lente1c = hc + rc1 * np.sin(theta1c)
254
255 x_lente2c = xc + dcx + rc2 * (1 - np.cos(theta2c))
256 y_lente2c = hc + rc2 * np.sin(theta2c)
257
258 plt.plot(x_lente1c, y_lente1c, color='g')
259 plt.plot(x_lente2c, y_lente2c, color='g')
260
261 foc=113.3
262 # Aggiungi sorgente e lente al grafico
263 plt.scatter([foc], [0], color='r', label='Punto focale')
264 plt.axvline(x=-100, color='g', linestyle='--', label='Bordi delle lenti
    convesse')
265 plt.axvline(x=-100, color='c', linestyle='--', label='Bordi della lente
    concava')
266 plt.axvline(x=0, color='y', linestyle=':', label='Sorgente di raggi
    paralleli')
267
268
269 # Configurazione del grafico
270 plt.xlim(-1 +10, xc + dcx + 100 -10)
271 plt.ylim(-altezza_massima, altezza_massima)
272 plt.xlabel('Asse Z (mm)')
273 plt.ylabel('Asse X (mm)')
274 plt.title('Ray Tracing 2D: Raggi Parallelari e Tripletto di Cooke')
275 plt.axvline(x=40.49, color='c', linestyle=':', label='Secondo piano
    principale (z=0 mm)', alpha=0.7)
276 plt.legend()
277 plt.grid(True)
278 plt.plot([xa + ra1 * (1 - np.cos(np.arcsin((ya/2)/ra1))), (xa + dax +
    ra2 * (1 - np.cos(np.arcsin((ya/2)/ra2))))], [ya/2, ya/2], color='g')
279 plt.plot([xa + ra1 * (1 - np.cos(np.arcsin((ya/2)/ra1))), (xa + dax +
    ra2 * (1 - np.cos(np.arcsin((ya/2)/ra2))))], [-ya/2, -ya/2], color='g
    ')
280 plt.plot([xb + rb1 * (1 - np.cos(np.arcsin((yb/2)/rb1))), (xb + dbx +
    rb2 * (1 - np.cos(np.arcsin((yb/2)/rb2))))], [yb/2, yb/2], color='c')
281 plt.plot([xb + rb1 * (1 - np.cos(np.arcsin((yb/2)/rb1))), (xb + dbx +
    rb2 * (1 - np.cos(np.arcsin((yb/2)/rb2))))], [-yb/2, -yb/2], color='c
    ')
282 plt.plot([xc + rc1 * (1 - np.cos(np.arcsin((yc/2)/rc1))), (xc + dcx +
    rc2 * (1 - np.cos(np.arcsin((yc/2)/rc2))))], [yc/2, yc/2], color='g')
283 plt.plot([xc + rc1 * (1 - np.cos(np.arcsin((yc/2)/rc1))), (xc + dcx +
    rc2 * (1 - np.cos(np.arcsin((yc/2)/rc2))))], [-yc/2, -yc/2], color='g
    ')
284 # Mostra il grafico con tutti i raggi e i bordi della lente
285
286
287 # Ottieni i valori originali delle etichette sull'asse X
288 xticks_originali = plt.gca().get_xticks()

```

```

289
290 # Sposta le etichette di 20 mm indietro
291 xticks_modificati = xticks_originali - 40.49
292
293
294 # Arrotonda le etichette a 2 cifre decimali
295 xticks_modificati = np.round(xticks_modificati, 2)
296
297 # Aggiorna le etichette dell'asse X con il nuovo valore
298 plt.xticks(xticks_originali, labels=xticks_modificati)
299
300 plt.show()
301
302 # Da qui inizia il grafico ingrandimento
303 # Consisterà in una replica del grafico precedente per determinare un
304 # grafico ingrandito
305 # Inizia il grafico ingrandimento (una sola figura)
306 plt.figure(figsize=(10, 5))
307
308 i=10
309 k=10
310 colormap = plt.cm.get_cmap('viridis', len(raggi_iniziali))
311 for raggio in raggi_iniziali:
312     if(5<k<=10):
313         i=i-1
314         k=k-1
315     elif(k==5):
316         i=5
317         k=k-1
318     else:
319         i=i+1
320         k=k-1
321     #sono interessato a trovare i punti di intersezione per la retta:
322     qa1=intercetta(0, raggio[0], np.tan(raggio[1])/n0)
323     intersezioni_a1 = punti_intersezione(np.tan(raggio[1])/n0, qa1, xa+
324                                         ra1, ha, ra1)
325     #calcolo del raggio dopo lo spazio libero
326     raggio_d[0] = calcolo_propagazione(raggio[0], raggio[1],
327                                         intersezioni_a1[1][0], n0)
328     raggio_d[1] = raggio[1]
329     # Calcolo del raggio dopo la prima lente
330     raggio_a1[0] = raggio_d[0]
331     raggio_a1[1] = calcolo_rifrazione(raggio_d[0], raggio_d[1], n0, n1,
332                                         ra1)
333     qa2=intercetta(intersezioni_a1[1][0], raggio_a1[0], np.tan(raggio_a1
334                                         [1])/n1)
335     intersezioni_a2 = punti_intersezione(np.tan(raggio_a1[1])/n1, qa2,
336                                         xa+dax+ra2, ha, ra2)
337     # Calcolo del raggio dopo lo spazio nel vetro

```

```

333 raggio_av[0] = calcolo_propagazione(raggio_a1[0], raggio_a1[1],
334     intersezioni_a2[0][0]-intersezioni_a1[1][0], n1)
335 raggio_av[1] = raggio_a1[1]
336 # Calcolo del raggio dopo la seconda lente
337 raggio_a2[0] = raggio_av[0]
338 raggio_a2[1] = calcolo_rifrazione(raggio_av[0], raggio_av[1], n1, n0
339     , ra2)
340 qb1=intercetta(intersezioni_a2[0][0], raggio_a2[0], np.tan(raggio_a2
341     [1])/n0)
342 intersezioni_b1 = punti_intersezione(np.tan(raggio_a2[1])/n0, qb1,
343     xb+rb1, hb, rb1)
344 #calcolo della propagazione nel vuoto dopo la seconda lente
345 raggio_e[0] = calcolo_propagazione(raggio_a2[0], raggio_a2[1],
346     intersezioni_b1[0][0]-intersezioni_a2[0][0] , n0)
347 raggio_e[1] = raggio_a2[1]
348 # Calcolo del raggio dopo la prima lente
349 raggio_b1[0] = raggio_e[0]
350 raggio_b1[1] = calcolo_rifrazione(raggio_e[0], raggio_e[1], n0, n2,
351     rb1)
352 qb2=intercetta(intersezioni_b1[0][0], raggio_b1[0], np.tan(raggio_b1
353     [1])/n2)
354 intersezioni_b2 = punti_intersezione(np.tan(raggio_b1[1])/n2, qb2,
355     xb+dbx+rb2, hb, rb2)
356 # Calcolo del raggio dopo lo spazio nel vetro
357 raggio_bv[0] = calcolo_propagazione(raggio_b1[0], raggio_b1[1],
358     intersezioni_b2[1][0]-intersezioni_b1[0][0] , n2)
359 raggio_bv[1] = raggio_b1[1]
360 # Calcolo del raggio dopo la seconda lente
361 raggio_b2[0] = raggio_bv[0]
362 raggio_b2[1] = calcolo_rifrazione(raggio_bv[0], raggio_bv[1], n2, n0
363     , rb2)
364 qc1=intercetta(intersezioni_b2[1][0], raggio_b2[0], np.tan(raggio_b2
365     [1])/n0)
366 intersezioni_c1 = punti_intersezione(np.tan(raggio_b2[1])/n0, qc1,
367     xc+rc1, hc, rc1)
368 # calcolo della propagazione nel vuoto dopo la seconda lente
369 raggio_f[0] = calcolo_propagazione(raggio_b2[0], raggio_b2[1],
370     intersezioni_c1[1][0]-intersezioni_b2[1][0] , n0)
371 raggio_f[1] = raggio_b2[1]
372 # Calcolo del raggio dopo la prima lente
373 raggio_c1[0] = raggio_f[0]
374 raggio_c1[1] = calcolo_rifrazione(raggio_f[0], raggio_f[1], n0, n1,
375     rc1)
376 qc2=intercetta(intersezioni_c1[1][0], raggio_c1[0], np.tan(raggio_c1
377     [1])/n1)
378 intersezioni_c2 = punti_intersezione(np.tan(raggio_c1[1])/n1, qc2,
379     xc+dcx+rc2, hc, rc2)
380 # Calcolo del raggio dopo lo spazio nel vetro
381 raggio_cv[0] = calcolo_propagazione(raggio_c1[0], raggio_c1[1],
382     intersezioni_c2[0][0]-intersezioni_c1[1][0] , n1)

```

```

366     raggio_cv[1] = raggio_c1[1]
367     # Calcolo del raggio dopo la seconda lente
368     raggio_c2[0] = raggio_cv[0]
369     raggio_c2[1] = calcolo_rifrazione(raggio_cv[0], raggio_cv[1], n1, n0
370         , rc2)
371     # calcolo della propagazione nel vuoto dopo la seconda lente
372     raggio_g[0] = calcolo_propagazione(raggio_c2[0], raggio_c2[1], 100+2
373         , n0)
374     raggio_g[1] = raggio_c2[1]
375     # Visualizzazione del percorso del raggio
376     plt.plot([intersezioni_c2[0][0], intersezioni_c2[0][0] + 100 + 2], [
377         raggio_cv[0], raggio_g[0]], color=colormap(i), alpha=0.7)

378 # Configurazione del grafico
379 plt.xlim(foc-0.65, foc+0.65)
380 plt.ylim(-0.05, 0.05)
381 plt.xlabel('Asse Z (mm)')
382 plt.ylabel('Asse X (mm)')
383 plt.title('Ray Tracing 2D: Zoom dell\'intersezione con asse ottico per
384     Tripletto di Cooke')
385 plt.scatter([foc], [0], color='r', label='Punto focale', alpha=0.7)
386 plt.axvline(x=foc, color='r', linestyle=':', label='Piano focale', alpha
387         =0.7)
388 plt.legend()
389 plt.grid(True)
390 # Mostra il grafico con tutti i raggi e i bordi della lente
391
392 # Ottieni i valori originali delle etichette sull'asse X
393 xticks_originali = plt.gca().get_xticks()
394
395 # Sposta le etichette di 20 mm indietro
396 xticks_modificati = xticks_originali - 40.49
397
398 # Arrotonda le etichette a 2 cifre decimali
399 xticks_modificati = np.round(xticks_modificati, 2)
400
401 # Aggiorna le etichette dell'asse X con il nuovo valore
402 plt.xticks(xticks_originali, labels=xticks_modificati)
403
404 plt.show()

405 foc=113.3

406 #grafico generico con punti generici
407 intersezioni_y=[]
408 # Parametri
409 ymin = -ya/2    # Limite inferiore dell'intervallo
410 ymax = ya/2    # Limite superiore dell'intervallo

```

```

411 n_punti = 50000 # Numero di punti da generare
412
413 # Genera punti casuali uniformemente distribuiti nell'intervallo [a, b]
414 puntiy = np.random.uniform(low=ymin, high=ymax, size=n_punti)
415
416 # Parametri
417 xmin = -ya/2 # Limite inferiore dell'intervallo
418 xmax = ya/2 # Limite superiore dell'intervallo
419
420
421 # Genera punti casuali uniformemente distribuiti nell'intervallo [a, b]
422 puntix = np.random.uniform(low=xmin, high=xmax, size=n_punti)
423
424 rxy = np.zeros(n_punti)
425 angxy = np.zeros(n_punti)
426 px = np.zeros(n_punti)
427 py = np.zeros(n_punti)
428
429 # Ciclo per calcolare rxy e l'angolo in gradi
430 for l in range(n_punti):
431     angxy[l] = np.degrees(np.arctan(puntiy[l] / puntix[l])) # Angolo in
        gradi
432     rxy[l] = np.sign(angxy[l])*np.sqrt(puntix[l]**2 + puntiy[l]**2) # Distanza
433     #plt.plot([0, puntix[l]], [0, puntiy[l]], color='g')
434     #print(puntix[l], puntiy[l], rxy, angxy)
435     l=l+1
436 plt.scatter(puntix, puntiy, color='y', marker='.', alpha=0.5)
437
438 plt.show()
439
440
441 n_raggi = n_punti # Numero di raggi paralleli
442
443 # Genera n_raggi paralleli orizzontali con diverse altezze
444 raggi_iniziali = np.array([[h, 0] for h in puntiy])
445
446 # Inizia il grafico (una sola figura)
447 plt.figure(figsize=(10, 5))
448
449 i=-1
450 colormap = plt.cm.get_cmap('viridis', n_punti)
451 for raggio in raggi_iniziali:
452     i=i+1
453
454 #sono interessato a trovare i punti di intersezione per la retta:
455     qa1=intercetta(0, raggio[0], np.tan(raggio[1])/n0)
456     intersezioni_a1 = punti_intersezione(np.tan(raggio[1])/n0, qa1, xa+r1
        , ha, ra1)
457     #calcolo del raggio dopo lo spazio libero

```

```

458 raggio_d[0] = calcolo_propagazione(raggio[0], raggio[1],
459     intersezioni_a1[1][0], n0)
460 raggio_d[1] = raggio[1]
461 # Calcolo del raggio dopo la prima lente
462 raggio_a1[0] = raggio_d[0]
463 raggio_a1[1] = calcolo_rifrazione(raggio_d[0], raggio_d[1], n0, n1,
464     ra1)
465 qa2=intercetta(intersezioni_a1[1][0], raggio_a1[0], np.tan(raggio_a1
466     [1])/n1)
467 intersezioni_a2 = punti_intersezione(np.tan(raggio_a1[1])/n1, qa2, xa+
468     dax+ra2, ha, ra2)
469 # Calcolo del raggio dopo lo spazio nel vetro
470 raggio_av[0] = calcolo_propagazione(raggio_a1[0], raggio_a1[1],
471     intersezioni_a2[0][0]-intersezioni_a1[1][0], n1)
472 raggio_av[1] = raggio_a1[1]
473 # Calcolo del raggio dopo la seconda lente
474 raggio_a2[0] = raggio_av[0]
475 raggio_a2[1] = calcolo_rifrazione(raggio_av[0], raggio_av[1], n1, n0,
476     ra2)
477 qb1=intercetta(intersezioni_a2[0][0], raggio_a2[0], np.tan(raggio_a2
478     [1])/n0)
479 intersezioni_b1 = punti_intersezione(np.tan(raggio_a2[1])/n0, qb1, xb+
480     rb1, hb, rb1)
481 #calcolo della propagazione nel vuoto dopo la seconda lente
482 raggio_e[0] = calcolo_propagazione(raggio_a2[0], raggio_a2[1],
483     intersezioni_b1[0][0]-intersezioni_a2[0][0] , n0)
484 raggio_e[1] = raggio_a2[1]
485 # Calcolo del raggio dopo la prima lente
486 raggio_b1[0] = raggio_e[0]
487 raggio_b1[1] = calcolo_rifrazione(raggio_e[0], raggio_e[1], n0, n2,
488     rb1)
489 qb2=intercetta(intersezioni_b1[0][0], raggio_b1[0], np.tan(raggio_b1
490     [1])/n2)
491 intersezioni_b2 = punti_intersezione(np.tan(raggio_b1[1])/n2, qb2, xb+
492     dbx+rb2, hb, rb2)
493 # Calcolo del raggio dopo lo spazio nel vetro
494 raggio_bv[0] = calcolo_propagazione(raggio_b1[0], raggio_b1[1],
495     intersezioni_b2[1][0]-intersezioni_b1[0][0] , n2)
496 raggio_bv[1] = raggio_b1[1]
497 # Calcolo del raggio dopo la seconda lente
498 raggio_b2[0] = raggio_bv[0]
499 raggio_b2[1] = calcolo_rifrazione(raggio_bv[0], raggio_bv[1], n2, n0,
500     rb2)
501 qc1=intercetta(intersezioni_b2[1][0], raggio_b2[0], np.tan(raggio_b2
502     [1])/n0)
503 intersezioni_c1 = punti_intersezione(np.tan(raggio_b2[1])/n0, qc1, xc+
504     rc1, hc, rc1)
505 # calcolo della propagazione nel vuoto dopo la seconda lente
506 raggio_f[0] = calcolo_propagazione(raggio_b2[0], raggio_b2[1],
507     intersezioni_c1[1][0]-intersezioni_b2[1][0] , n0)

```

```

491 raggio_f[1] = raggio_b2[1]
492 # Calcolo del raggio dopo la prima lente
493 raggio_c1[0] = raggio_f[0]
494 raggio_c1[1] = calcolo_rifrazione(raggio_f[0], raggio_f[1], n0, n1,
495   rc1)
496 qc2=intercetta(intersezioni_c1[1][0], raggio_c1[0], np.tan(raggio_c1
497   [1])/n1)
498 intersezioni_c2 = punti_intersezione(np.tan(raggio_c1[1])/n1, qc2, xc+
499   dcx+rc2, hc, rc2)
500 # Calcolo del raggio dopo lo spazio nel vetro
501 raggio_cv[0] = calcolo_propagazione(raggio_c1[0], raggio_c1[1],
502   intersezioni_c2[0][0]-intersezioni_c1[1][0], n1)
503 raggio_cv[1] = raggio_c1[1]
504 # Calcolo del raggio dopo la seconda lente
505 raggio_c2[0] = raggio_cv[0]
506 raggio_c2[1] = calcolo_rifrazione(raggio_cv[0], raggio_cv[1], n1, n0,
507   rc2)
508 # calcolo della propagazione nel vuoto dopo la seconda lente
509 raggio_g[0] = calcolo_propagazione(raggio_c2[0], raggio_c2[1], 100+2 ,
510   n0)
511 raggio_g[1] = raggio_c2[1]

512 #grafico le intersezioni con y=0
513 #print(f"valori di intersezione:{raggioc2[0]/raggio_dopo_vetroc[1]}")
514 m_c = raggio_g[1] # Pendenza dopo la lente
515 y_c = raggio_c2[0] # Altezza dopo la lente
516 y_intersezione = m_c * (foc - intersezioni_c2[0][0]) + y_c
517 #print(f"inclinazione raggio {i},{m_c}, altezza iniziale:{y_c},
518   posizione risultante: {y_intersezione}")
519 intersezioni_y.append(y_intersezione)

520 plt.figure(figsize=(5, 5))
521 for l in range(n_punti):
522   px[l]=intersezioni_y[l]*np.cos(angxy[l]) # Distanza x
523   py[l]=intersezioni_y[l]*np.sin(angxy[l]) # Distanza y
524   l=l+1
525 plt.scatter(px, py, color='y', marker='.', alpha=0.3, s=20)
526 plt.scatter(0, 0, color='r', marker='.', alpha=0.7, label='Asse ottico')
527 plt.xlabel('Asse Y (mm)')
528 plt.ylabel('Asse X (mm)')
529 plt.title('Distribuzione dei punti di intersezione dei raggi con
530   l asse focale \n per il tripletto di Cooke')
531 plt.legend()
532 plt.show()

533 #intersezioni per lente spessa
534 plt.figure(figsize=(5, 5))
535 for l in range(n_punti):
536   pxx[l]=intersezioni_yy[l]*np.cos(angxyy[l]) # Distanza x
537   pyy[l]=intersezioni_yy[l]*np.sin(angxyy[l]) # Distanza y

```

```

533     l=l+1
534     plt.scatter(pxx, pyy, color='b', marker='.', alpha=0.3, s=20)
535     plt.scatter(0, 0, color='r', marker='.', alpha=0.7, label='Asse ottico')
536     plt.xlabel('Asse Y (mm)')
537     plt.ylabel('Asse X (mm)')
538     plt.title('Distribuzione dei punti di intersezione dei raggi con
539         l asse focale \n per la lente spessa')
540     plt.legend()
541     plt.show()
542
543 plt.figure(figsize=(5, 5))
544 for l in range(n_punti):
545     pxx[l]=intersezioni_yy[l]*np.cos(angxyy[l]) # Distanza x
546     pyy[l]=intersezioni_yy[l]*np.sin(angxyy[l]) # Distanza y
547     px[l]=intersezioni_y[l]*np.cos(angxy[l]) # Distanza x
548     py[l]=intersezioni_y[l]*np.sin(angxy[l]) # Distanza y
549     l=l+1
550     plt.scatter(pxx, pyy, color='b', marker='.', alpha=0.3, s=20)
551     plt.scatter(px, py, color='y', marker='.', alpha=0.3, s=20)
552     plt.scatter(0, 0, color='r', marker='.', alpha=0.7, label='Asse ottico')
553     plt.xlabel('Asse Y (mm)')
554     plt.ylabel('Asse X (mm)')
555     plt.title('Distribuzione dei punti di intersezione dei raggi con
556         l asse focale \n per il tripletto di Cooke e la lente spessa')
557     plt.legend()
558     plt.show()
559
560 distanze = np.sqrt(px**2 + py**2)
561
562 # Definire l'ampiezza dello spessore dx (ad esempio 1.0)
563 dx = 0.0000007
564
565 # Definire i bin (intervalli di distanza)
566 bins = np.arange(0, np.max(distanze) + dx, dx)
567
568 # Contare il numero di punti in ogni bin (istogramma delle distanze)
569 conteggi, bin_edges = np.histogram(distanze, bins=bins)
570
571 # Calcolare il centro di ogni bin per la rappresentazione grafica
572 centro_bins = (bin_edges[:-1] + bin_edges[1:]) / 2
573
574 # Calcolare l'area della corona circolare per ogni bin
575 aree_corone = np.pi * (bin_edges[1]**2 - bin_edges[:-1]**2)
576
577 # Normalizzare i conteggi per l'area della corona corrispondente
578 conteggi_normalizzati = conteggi / aree_corone
579
580 # Trova il massimo dell'istogramma

```

```

581 max_conteggi = np.max(conteggi_normalizzati)
582
583 # Calcola la mezza altezza
584 mezza_altezza = max_conteggi / 2
585
586 # Trova gli indici a sinistra e destra della mezza altezza
587 left_idx = np.where(conteggi_normalizzati >= mezza_altezza)[0][0]
588 right_idx = np.where(conteggi_normalizzati <= mezza_altezza)[0][-1]
589
590 # Calcola l'ampiezza a mezza altezza (FWHM)
591 FWHM = centro_bins[right_idx] - centro_bins[left_idx]
592
593 print(f"Ampiezza a mezza altezza (FWHM): {FWHM} mm")
594 print(f"Valore della mezza altezza: {mezza_altezza:.3f}")
595
596 # Creare il grafico
597 plt.plot(centro_bins, conteggi_normalizzati, marker='o', linestyle='--',
      color='y')
598
599 # Imposta i limiti sull'asse x per ridurre la lunghezza
600 plt.xlim(0, 0.0000007)
601
602 # Etichette e titolo
603 plt.xlabel('Distanza dall\'origine (mm)')
604 plt.ylabel('Numero di punti per unit di area della corona (1/mm )')
605 plt.title(f'Distribuzione dei punti di intersezione fra i raggi e il
      piano focale normalizzati per l\'area della corona circolare
      considerata')
606
607 # Mostra il grafico
608 plt.grid(True)
609 plt.show()
610
611 distanze1 = np.sqrt(pxx**2 + pyy**2)
612
613 # Definire l'ampiezza dello spessore dx (ad esempio 1.0)
614 dx = 0.000000007
615
616 # Definire i bin (intervalli di distanza)
617 bins1 = np.arange(0, np.max(distanze1) + dx, dx)
618
619 # Contare il numero di punti in ogni bin (istogramma delle distanze)
620 conteggi, bin_edges = np.histogram(distanze1, bins=bins1)
621
622 # Calcolare il centro di ogni bin per la rappresentazione grafica
623 centro_bins = (bin_edges[:-1] + bin_edges[1:]) / 2
624
625
626 # Calcolare l'area della corona circolare per ogni bin
627 aree_corone = np.pi * (bin_edges[1]**2 - bin_edges[:-1]**2)

```

```

628
629 # Normalizzare i conteggi per l'area della corona corrispondente
630 conteggi_normalizzati = conteggi / aree_corone
631
632 # Trova il massimo dell'istogramma
633 max_conteggi = np.max(conteggi_normalizzati)
634
635 # Calcola la mezza altezza
636 mezza_altezza = max_conteggi / 2
637
638 # Trova gli indici a sinistra e destra della mezza altezza
639
640 left_idx = np.where(conteggi_normalizzati >= mezza_altezza)[0][0]
641 right_idx = np.where(conteggi_normalizzati <= mezza_altezza)[0][-1]
642
643
644 print(f"Indice sinistro (left_idx): {left_idx}")
645 print(f"Indice destro (right_idx): {right_idx}")
646 print(f"Centro bins left: {centro_bins[left_idx]}")
647 print(f"Centro bins right: {centro_bins[right_idx]}")
648
649 # Verifica se gli indici sono uguali
650 if left_idx == right_idx:
651     print("Gli indici sono uguali, FWHM      zero.")
652 else:
653     # Calcola l'ampiezza a mezza altezza (FWHM)
654     FWHM = centro_bins[right_idx] - centro_bins[left_idx]
655     print(f"Ampiezza a mezza altezza (FWHM): {FWHM} mm")
656
657
658 # Calcola l'ampiezza a mezza altezza (FWHM)
659 FWHM = centro_bins[right_idx] - centro_bins[left_idx]
660
661 print(f"Ampiezza a mezza altezza (FWHM): {FWHM} mm")
662 print(f"Valore della mezza altezza: {mezza_altezza:.3f}")
663
664 # Creare il grafico
665 plt.plot(centro_bins, conteggi_normalizzati, marker='s', linestyle='--',
666           color='b', markersize=1)
667
668 # Creare il grafico con linea della mezza altezza
669 #plt.plot(centro_bins, conteggi_normalizzati, marker='s', linestyle='--',
670 #          color='r')
671 #plt.axhline(y=mezza_altezza, color='r', linestyle='--', label='Mezza
672 #              Altezza')
673
674 # Imposta i limiti sull'asse x per ridurre la lunghezza
675 plt.xlim(0, 0.0000007)
676
677 # Etichette e titolo

```

```

675 plt.xlabel('Distanza dall\'origine (mm)')
676 plt.ylabel('Numero di punti per unità di area della corona (1/mm²)')
677 plt.title(f'Distribuzione dei punti di intersezione fra i raggi e il piano focale normalizzati per l\'area della corona circolare considerata')
678
679
680 # Mostra il grafico
681 plt.grid(True)
682 plt.show()
683
684
685 # Calcolo normalizzato
686
687 # Calcolo delle distanze per entrambi i set
distanze = np.sqrt(px**2 + py**2)
distanze1 = np.sqrt(pxx**2 + pyy**2)
688
689
690
691
692 # Definire l'ampiezza dello spessore dx (ad esempio 0.005)
dx = 0.000000007
693
694
695 # Definire i bin (intervalli di distanza), utilizzando il massimo delle distanze combinate
bins = np.arange(0, max(np.max(distanze), np.max(distanze1)) + dx, dx)
696
697
698 # Contare il numero di punti in ogni bin per entrambi i set di dati
conteggi, bin_edges = np.histogram(distanze, bins=bins)
conteggi1, bin_edges1 = np.histogram(distanze1, bins=bins)
699
700
701 # Calcolare il centro di ogni bin per la rappresentazione grafica
centro_bins = (bin_edges[:-1] + bin_edges[1:]) / 2
702
703
704 # Calcolare l'area della corona circolare per ogni bin
aree_corone = np.pi * (bin_edges[1]**2 - bin_edges[:-1]**2)
705
706
707 # Normalizzare i conteggi per l'area della corona corrispondente
conteggi_normalizzati = conteggi / aree_corone
conteggi1_normalizzati = conteggi1 / aree_corone
708
709
710
711 # Creare il grafico sovrapposto con i conteggi normalizzati
plt.plot(centro_bins, conteggi1_normalizzati, marker='o', linestyle='--',
color='y', markersize=3, label='Tripletto di Cooke')
712 plt.plot(centro_bins, conteggi_normalizzati, marker='s', linestyle='--',
color='b', markersize=3, label='Lente spessa')
713
714
715
716
717
718 # Imposta i limiti sull'asse x per ridurre la lunghezza
plt.xlim(0, 0.0000007)
719
```

```

720
721
722 # Etichette e titolo
723 plt.xlabel('Distanza dall\'origine (mm)')
724 plt.ylabel('Numero di punti per unit di area della corona (1/ mm )')
725 plt.title(f'Confronto fra le distribuzioni dei punti di intersezione fra
726 i raggi e il piano focale \n normalizzati per l\'area della corona
727 circolare considerata')
728
729 # Aggiungere una legenda per distinguere i due set
730 plt.legend()
731
732 # Mostra il grafico
733 plt.grid(True)
734 plt.show()
735
736 # Calcolo delle distanze per entrambi i set
737 distanze = np.sqrt(px**2 + py**2)
738 distanze1 = np.sqrt(pxx**2 + pyy**2)
739
740 # Definire l'intervallo di interesse per le distanze (es. tra 0 e
741 # 0.0000007 mm)
742 limite_inferiore = 0
743 limite_superiore = 0.000001
744
745 # Filtrare i dati per ottenere solo quelli nell'intervallo desiderato
746 distanze_filtrate = distanze[(distanze >= limite_inferiore) & (distanze
747 <= limite_superiore)]
748 distanze1_filtrate = distanze1[(distanze1 >= limite_inferiore) & (
749 distanze1 <= limite_superiore)]
750
751 # Definire l'ampiezza dello spessore dx
752 dx = 0.0000001
753
754 # Definire i bin (intervalli di distanza) limitati ai dati filtrati
755 bins = np.arange(limite_inferiore, limite_superiore + dx, dx)
756
757 # Contare il numero di punti in ogni bin per i dati filtrati
758 conteggi, bin_edges = np.histogram(distanze_filtrate, bins=bins)
759 conteggi1, bin_edges1 = np.histogram(distanze1_filtrate, bins=bins)
760
761 # Calcolare il centro di ogni bin per la rappresentazione grafica
762 centro_bins = (bin_edges[:-1] + bin_edges[1:]) / 2
763
764 # Calcolare l'area della corona circolare per ogni bin
765 aree_corone = np.pi * (bin_edges[1]**2 - bin_edges[:-1]**2)
766
767 # Normalizzare i conteggi per l'area della corona corrispondente
768 conteggi_normalizzati = conteggi / aree_corone

```

```

765 conteggi1_normalizzati = conteggi1 / aree_corone
766
767 # Creare il grafico sovrapposto con i conteggi normalizzati
768 plt.plot(centro_bins, conteggi1_normalizzati, marker='o', linestyle='--',
    color='y', markersize=3, label='Tripletto di Cooke')
769 plt.plot(centro_bins, conteggi1_normalizzati, marker='s', linestyle='--',
    color='b', markersize=3, label='Lente spessa')
770
771 # Etichette e titolo
772 plt.xlabel('Distanza dall\'origine (mm)')
773 plt.ylabel('Numero di punti per unit di area della corona (1/mm )')
774 plt.title(f'Confronto fra le distribuzioni dei punti di intersezione fra
    i raggi e il piano focale \n normalizzati per l\'area della corona
    circolare considerata')
775
776 # Aggiungere una legenda per distinguere i due set
777 plt.legend()
778
779 # Mostra il grafico
780 plt.grid(True)
781 plt.show()

```

A.2 Rappresentazione matriciale del tripletto di cooke

$$\begin{aligned}
 & \left(\frac{\begin{vmatrix} n_0 & n_1 \\ n_1 & da \end{vmatrix} - n_0 n_1}{n_1 R_{a1}} + \frac{n_0 n_1}{n_0 R_{a2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_2 \\ n_2 & dc \end{vmatrix} - n_0 n_2}{n_2 R_{c1}} + \frac{n_0 n_2}{n_0 R_{c2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_2 \\ n_2 & da \end{vmatrix} - n_0 n_2}{n_2 R_{d1}} + \frac{n_0 n_2}{n_0 R_{d2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_1 \\ n_1 & da \end{vmatrix} - n_0 n_1}{n_1 R_{a1}} + \frac{n_0 n_1}{n_0 R_{a2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_2 \\ n_2 & dc \end{vmatrix} - n_0 n_2}{n_2 R_{c1}} + \frac{n_0 n_2}{n_0 R_{c2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_2 \\ n_2 & da \end{vmatrix} - n_0 n_2}{n_2 R_{d1}} + \frac{n_0 n_2}{n_0 R_{d2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_1 \\ n_1 & da \end{vmatrix} - n_0 n_1}{n_1 R_{a1}} + \frac{n_0 n_1}{n_0 R_{a2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_2 \\ n_2 & dc \end{vmatrix} - n_0 n_2}{n_2 R_{c1}} + \frac{n_0 n_2}{n_0 R_{c2}} \right) \cdot \\
 & \left(\frac{\begin{vmatrix} n_0 & n_2 \\ n_2 & da \end{vmatrix} - n_0 n_2}{n_2 R_{d1}} + \frac{n_0 n_2}{n_0 R_{d2}} \right) \cdot
 \end{aligned}$$

$$\begin{aligned}
& \left[\frac{1, \frac{dc}{n_2^2} n_0 n_2}{n_2^2 R_{c1}} \left(\left| 1, \frac{da}{n_1^2} n_0 n_1 \right| \right) \right] \\
& + \frac{\frac{dc}{n_2} n_0 \left| 1, \frac{da}{n_1^2} n_0 n_1 \right|}{n_2^2 R_{c1}} \\
& + \left(1 + \frac{da}{n_1^2} n_0 n_1 \right) \left(\left(\frac{n_0 n_2}{n_0^2}, \frac{dc}{n_2} n_0 n_2 \right) \right. \\
& \quad \left. \left(\frac{n_0 n_2}{n_2 R_{c1}}, \frac{dc}{n_2} n_0 n_2 \right) + \frac{n_0 n_2}{n_0 R_{c2}} \right) \\
& + \left(1 + \frac{dc}{n_2^2} n_0 n_2 \right) \left(1 + \frac{da}{n_1^2} n_0 n_1 \right) \\
& + \left(1 + \frac{dc}{n_2^2} n_0 n_2 \right) \left(1 + \frac{da}{n_1^2} n_0 n_1 \right) \\
& + \left(1 + \frac{da}{n_1^2} n_0 n_1 \right) \left(\left(\frac{n_0 n_2}{n_0^2}, \frac{dc}{n_2} n_0 n_2 \right) \right. \\
& \quad \left. \left(\frac{n_0 n_2}{n_2 R_{c1}}, \frac{dc}{n_2} n_0 n_2 \right) + \frac{n_0 n_2}{n_0 R_{c2}} \right) \\
& + \left(1 + \frac{dc}{n_2^2} n_0 n_2 \right) \left(\left(\frac{n_0 n_1}{n_0^2}, \frac{d}{n_1^2} n_0 n_1 \right) \right. \\
& \quad \left. \left(\frac{n_0 n_1}{n_2 R_{c1}}, \frac{d}{n_1^2} n_0 n_1 \right) + \frac{n_0 n_1}{n_0 R_{c2}} \right) \\
& + \left(1 + \frac{dc}{n_2^2} n_0 n_2 \right) \left(\left(\frac{n_0 n_1}{n_0^2}, \frac{d}{n_1^2} n_0 n_1 \right) \right. \\
& \quad \left. \left(\frac{n_0 n_1}{n_2 R_{c1}}, \frac{d}{n_1^2} n_0 n_1 \right) + \frac{n_0 n_1}{n_0 R_{c2}} \right)
\end{aligned}$$

$$\begin{aligned}
& \left[\begin{array}{c} \cdot \left(1, \text{dc } n_0 n_2 \right) \left(1, \text{da } n_0 n_1 \right) \\ \left(n_2^2 \text{Re}_1 \right) \end{array} \right] \left[\begin{array}{c} \cdot \left(1, \text{da } n_0 n_1 \right) \\ \left(n_1^2 \text{Re}_1 \right) \end{array} \right] \left[\begin{array}{c} \text{dc } n_0 \left(1, \text{da } n_0 n_1 \right) \\ \left(n_2^2 \text{Re}_1 \right) \end{array} \right] \\
& + \left[\begin{array}{c} \text{da } n_0 \left(\begin{array}{c} n_0 n_2 \\ n_0^2 \text{Re}_1 \end{array} \right) \end{array} \right] \left[\begin{array}{c} n_0 n_2 \left(\begin{array}{c} n_2^2 \text{dc } n_0 n_2 \\ n_0^2 n_2 \text{Re}_2 \end{array} \right) \\ , n_0 n_2 \left(\begin{array}{c} \text{da } n_0 \\ n_1^2 \text{Re}_1 \end{array} \right) \end{array} \right] \left[\begin{array}{c} \text{dd } \left(\begin{array}{c} 1, \text{da } n_0 n_1 \\ n_1^2 \text{Re}_1 \end{array} \right) \\ , \left(1, \text{dc } n_0 n_2 \right) \left(1, \text{da } n_0 n_1 \right) \end{array} \right] \\
& + \left[\begin{array}{c} n_0 n_1 \\ n_1 \text{Re}_2 \end{array} \right] \left[\begin{array}{c} -n_0 n_1 \\ n_0 \text{Re}_2 \end{array} \right] \left[\begin{array}{c} n_0 n_1 \left(\begin{array}{c} n_1 \text{da } (-n_0 n_1) \\ n_0 n_1 \text{Re}_2 \end{array} \right) \\ , -n_0 n_1 \end{array} \right] \\
& + \left[\begin{array}{c} \text{dc } n_0 \\ n_1 \text{Re}_1 \end{array} \right] \left[\begin{array}{c} n_0 n_1 \\ n_0^2 \text{Re}_2 \end{array} \right] \left[\begin{array}{c} n_0 n_1 \left(\begin{array}{c} n_1 \text{da } (-n_0 n_1) \\ n_0 n_1 \text{Re}_2 \end{array} \right) \\ , -n_0 n_1 \end{array} \right] \\
& + \left[\begin{array}{c} n_2^2 \\ n_1^2 \text{Re}_1 \end{array} \right] \left[\begin{array}{c} n_0 n_2 \left(\begin{array}{c} n_2^2 \text{dc } n_0 n_2 \\ n_0^2 n_2 \text{Re}_2 \end{array} \right) \\ , n_0 n_2 \left(\begin{array}{c} \text{da } n_0 \\ n_1^2 \text{Re}_1 \end{array} \right) \end{array} \right] \left[\begin{array}{c} \text{dd } \left(\begin{array}{c} 1, \text{da } n_0 n_1 \\ n_1^2 \text{Re}_1 \end{array} \right) \\ , \left(1, \text{dc } n_0 n_2 \right) \left(1, \text{da } n_0 n_1 \right) \end{array} \right] \\
& + \left[\begin{array}{c} \text{da } n_0 \\ n_0 \end{array} \right] \left[\begin{array}{c} n_0 n_2 \\ n_0^2 \text{Re}_1 \end{array} \right] \left[\begin{array}{c} n_0 n_2 \left(\begin{array}{c} n_2^2 \text{dc } n_0 n_2 \\ n_0^2 n_2 \text{Re}_2 \end{array} \right) \\ , n_0 n_2 \left(\begin{array}{c} \text{da } n_0 \\ n_1^2 \text{Re}_1 \end{array} \right) \end{array} \right] \left[\begin{array}{c} \text{dd } \left(\begin{array}{c} 1, \text{da } n_0 n_1 \\ n_1^2 \text{Re}_1 \end{array} \right) \\ , \left(1, \text{dc } n_0 n_2 \right) \left(1, \text{da } n_0 n_1 \right) \end{array} \right] \\
& + \left[\begin{array}{c} n_0 n_1 \\ n_1 \text{Re}_1 \end{array} \right] \left[\begin{array}{c} n_0 n_2 \\ n_0^2 \text{Re}_2 \end{array} \right] \left[\begin{array}{c} n_0 n_1 \left(\begin{array}{c} n_1 \text{da } (-n_0 n_1) \\ n_0 n_1 \text{Re}_2 \end{array} \right) \\ , -n_0 n_1 \end{array} \right] \\
& + \left[\begin{array}{c} n_0 \\ n_1 \text{Re}_2 \end{array} \right] \left[\begin{array}{c} n_0 n_1 \\ n_0^2 \text{Re}_1 \end{array} \right] \left[\begin{array}{c} n_0 n_1 \left(\begin{array}{c} n_1 \text{da } (-n_0 n_1) \\ n_0 n_1 \text{Re}_2 \end{array} \right) \\ , -n_0 n_1 \end{array} \right]
\end{aligned}$$

Ringraziamenti

Ognuno affronta un viaggio diverso durante il proprio percorso di studi, le incognite sono molte ed il rischio di perderti è elevato. È dunque fondamentale avere al proprio fianco persone capaci di correggerti nei momenti di distrazione e di illuminarti la strada quando fa buio. Il mio pensiero va dunque a chiunque abbia costituito per me un amico, un supporto nelle avversità e mi abbia corretto quando era il mio turno di imparare, i ringraziamenti di questa tesi sono dedicati a tutti voi.

Il primo pensiero va a Mamma e Papà. Li ringrazio per avermi aiutato ad essere la persona che sto diventando e per non essersi mai allontanati quando avevo bisogno di loro. In particolare ringrazio la mamma per avermi trasmesso la passione e lo spirito giocoso nei confronti della logica e della matematica. Ringrazio invece il papà per i consigli inestimabili, la disciplina e motivazione che mi ha condiviso, perchè: "il tepore delle lenzuola non ha mai fatto bollire le pentole".

I traguardi che ho raggiunto li devo a voi.

Ringrazio Fra e Marco per l'approccio più giovanile ma al contempo maturo che mi hanno insegnato, e per il burrascoso supporto e le risate che mi hanno sempre fornito.

Ringrazio la Nonna per tutte le candele accese durante gli esami, e per essere sempre stata la mia fan numero uno.

Ringrazio Albi, Simi, Ila e Argo per tutti i sorrisi regalati.

Ma soprattutto ringrazio la Zia per avermi insegnato di non smettere mai di sorridere nonostante tutte le difficoltà, grazie Biba, sei la ventata di aria fresca alla fine di una giornata interminabile.

Ringrazio infine i miei nonni per non aver mai dubitato di me.

In particolare ringrazio Alice (Fig. (23)), il mio raggio di Sole, per avermi aiutato a ritrovare il bambino dentro di me, permettendomi di scoprire la meraviglia dentro tutte le cose. La ringrazio perchè mi ha insegnato in un modo tutto suo a mettermi in discussione e imparare senza aver paura di farlo. Grazie Ali, mi rendi felice, mi hai fatto capire che una relazione ha molto più che da regalare che semplice affetto reciproco, e che è tanto facile aiutarsi e migliorarsi a vicenda.

Ci tengo a ringraziare Matte e Gio, ci siete stati sempre. Siete quegli amici che non hai bisogno di sentire sempre per sapere che ci sono, ma che quando ne hai bisogno, sono in prima fila al tuo fianco ad aiutarti a combattere le tue battaglie più dure. Grazie, vi voglio bene.

Ringrazio infine Seba, Steve, Pietro, Pier, Dani, Panca gli Stokes e tutti i miei amici e amiche, vicini e lontani, che mi hanno fatto divertire ed emozionare innumerevoli volte, siete stati la base dell'ambiente universitario che tutti sperano di avere. In particolare ringrazio Lorenzo e Matteneneo, i miei Topg, con cui ho condiviso felicemente sia i momenti di gioia che i momenti di disperazione.

In definitiva ci tengo a ringraziare ancora chiunque mi voglia bene e mi abbia regalato un sorriso.