



Optyka soczewek

Grupa projektowa: 1B

Wykonawcy:

*Jakub Stosik s204300
Karol Pluto Prądzynski s203561*

Wersja dokumentu: 1.3

15 XII 2025 r.

1 Wstęp

Optyka jest działem fizyki zajmującą się światłem - jego prawami, rozchodzeniem się i oddziaływaniem z materią. Ważnym zagadnieniem tego działu jest optyka geometryczna przedstawiająca opis załamywania się promieni światła w różnych ośrodkach. W tym projekcie zajmujemy się załamaniem promieni światła na soczewkach, które z fizycznego punktu widzenia są przezroczystymi obiektami o dwóch współosiowych powierzchniach załamujących światło.

2 Cel projektu

Celem projektu jest zaprojektowanie modelu fizycznego promienia światlnego przechodzącego przez soczewkę o zadanych parametrach.

3 Opis modelowanego zjawiska

Soczewki mogą wytwarzać obraz źródła światła zmieniając kierunek promieni światlnych wychodzących z jego powierzchni. Obraz powstaje gdy po zmianie kierunku przecinają się promienie (tworząc obraz rzeczywisty), lub ich przedłużenia (tworząc obraz pozorny).

3.1 Modelowanie soczewek

W naszym programie zaimplementowane zostaną 4 typy soczewek: dwuwypukłe, dwuwklęsłe, płasko-wypukłe, płasko-wklęsłe. Dla każdego z tych typów równania opisujące soczewki na płaszczyźnie XZ , równania wyglądają następująco:

$$(x - x_1)^2 + z^2 = R_1^2 \quad (1)$$

gdzie x_1 , R_1 są to parametry zadane przez użytkownika - środek oraz promień pierwszego okręgu.

$$(x - x_2)^2 + z^2 = R_2^2 \quad (2)$$

gdzie x_2 , R_2 są to parametry zadane przez użytkownika - środek oraz promień drugiego okręgu.

Warto zauważyć, że środki obu okręgów leżą na tej samej osi (prostej $z = z_1 = z_2 = 0$)

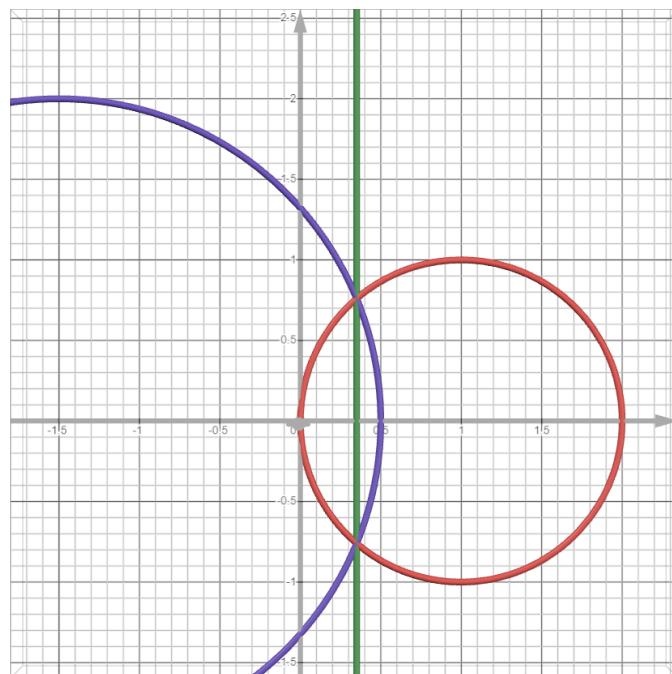
Rozwiążując te równania ze względu na x otrzymujemy prostą, która wyznaczy nam środek soczewki dwuwypukłej:

$$x = \frac{x_1^2 - x_2^2 - R_1^2 + R_2^2}{2(x_1 - x_2)} \quad (3)$$

Aby modelowanie soczewek za pomocą powyższych równań miało sens, musi zostać spełniony następujący warunek:

$$|R_1 - R_2| < |x_1 - x_2| < R_1 + R_2 \quad (4)$$

gdzie $R_1, R_2 > 0$ oraz $x_1 > x_2$



Rysunek 1: Matematyczna wizualizacja cienkiej soczewki dwuwypukłej

Soczewki dwuwklesłe

Dla soczewek dwuwkleskich, które matematycznie są opisane za pomocą dwóch okręgów na płaszczyźnie XZ , dwóch prostych ograniczających soczewkę z góry i

z dołu (apertura), równania wyglądają następująco:

$$(x - x_1)^2 + z^2 = R_1^2 \quad (5)$$

gdzie x_1, R_1 są to parametry zadane przez użytkownika - środek oraz promień pierwszego okręgu.

$$(x - x_2)^2 + z^2 = R_2^2 \quad (6)$$

gdzie x_2, R_2 są to parametry zadane przez użytkownika - środek oraz promień drugiego okręgu.

Warto zauważyć, że środki obu okręgów leżą na tej samej osi (prostej $z = z_1 = z_2 = 0$)

Rozwiązujejąc te równania ze względu na x - podobnie jak powyżej - otrzymujemy prostą, która wyznaczy nam środek soczewki dwuwkleszej:

$$x = \frac{x_1^2 - x_2^2 - R_1^2 + R_2^2}{2(x_1 - x_2)} \quad (7)$$

Proste ograniczające soczewkę z góry:

$$y = \pm a \quad (8)$$

gdzie $a < \min(|R_1|, |R_2|)$

Aby modelowanie soczewek za pomocą powyższych równań miało sens, musi zostać spełniony następujący warunek:

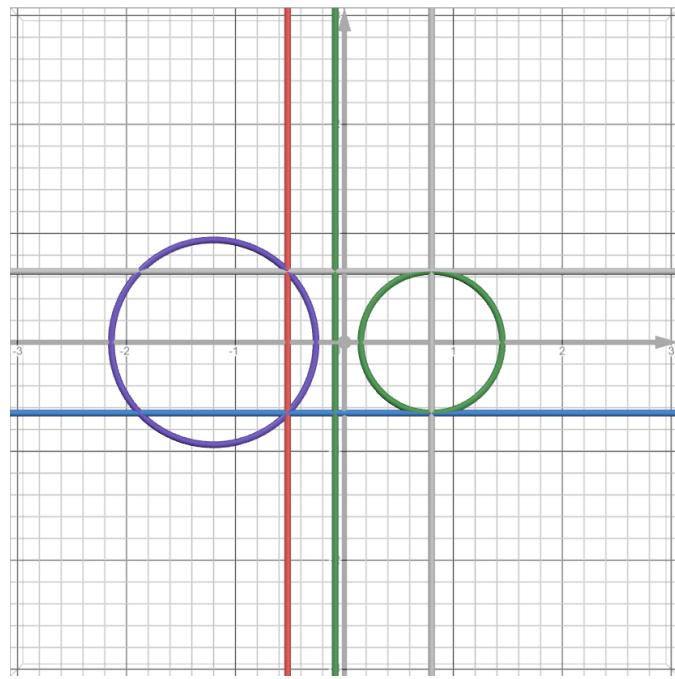
$$|x_1 - x_2| > R_1 + R_2 \quad (9)$$

gdzie $R_1, R_2 > 0$ oraz $x_1 > x_2$

Dodatkowo w programie podczas rysowania soczewek dwuwkleskich, będziemy je ograniczać z lewej jak i z prawej strony odpowiednimi równaniami:

$$x = x_2 + \sqrt{R_2^2 - a^2} \quad (10)$$

$$x = x_1 - \sqrt{R_1^2 - a^2} \quad (11)$$



Rysunek 2: Matematyczna wizualizacja cienkiej soczewki dwuwkleszej

Soczewki płasko-wypukłe

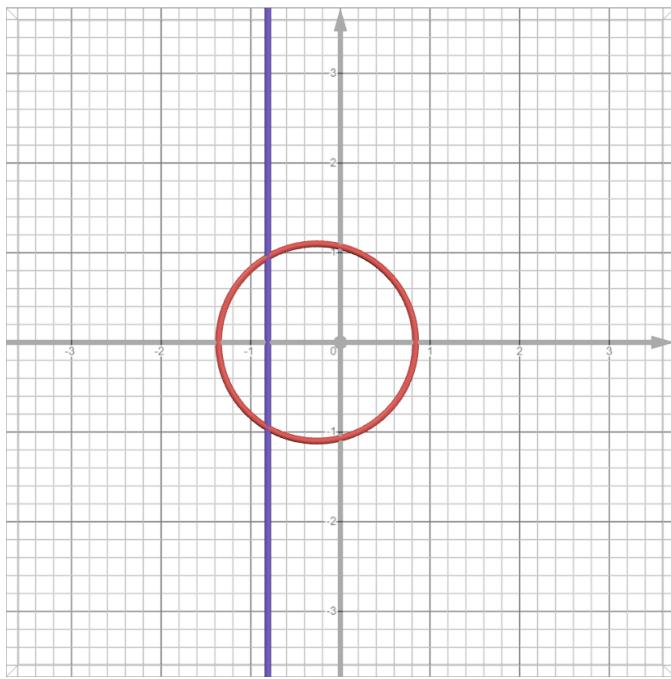
Dla soczewek płasko-wypukłych, które matematycznie są opisane za pomocą jednego okręgu na płaszczyźnie XZ oraz jednej prostej ograniczającej soczewkę (płaska jej część), równania wyglądają następująco:

$$(x - x_1)^2 + z^2 = R_1^2 \quad (12)$$

gdzie $x_1, R_1 > 0$ są to parametry zadane przez użytkownika - środek oraz promień okręgu.

$$x = a \quad (13)$$

gdzie $x_1 - R_1 < a < x_1 + R_1$



Rysunek 3: Matematyczna wizualizacja cienkiej soczewki płasko-wypukłej

Soczewki płasko-wklesłe

Dla soczewek płasko-wklesłych, które matematycznie są opisane za pomocą jednego okręgu na płaszczyźnie XZ , jedenj prostej ograniczającej soczewkę (płaska jej część) oraz dwóch prostych ograniczających soczewkę z góry i z dołu (apertura), równania wyglądają następująco:

$$(x - x_2)^2 + z^2 = R_2^2 \quad (14)$$

gdzie x_2 , R_2 są to parametry zadane przez użytkownika - środek oraz promień okręgu.

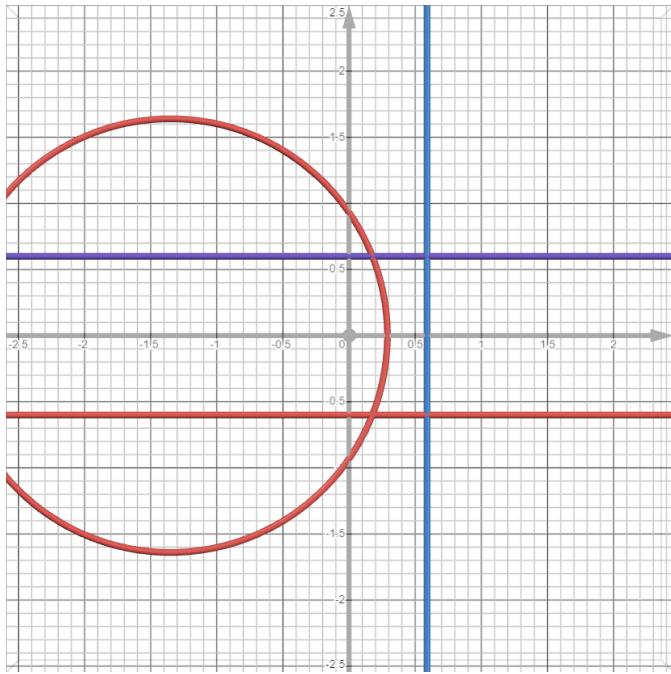
Proste ograniczające soczewkę z góry:

$$y = \pm a \quad (15)$$

gdzie $0 < a < R_2$

Prosta ograniczająca soczewkę

$$x = x_2 + R_2 + \frac{a}{2} \quad (16)$$



Rysunek 4: Matematyczna wizualizacja cienkiej soczewki płasko-wklęsłej

3.2 Rysowanie promienia świetlnego - ray tracing

Promień będziemy rysować w trzech etapach. Pierwszy etap to przebieg promienia od punktu startowego do punktu, w którym pada na powierzchnię soczewki z lewej strony, bo zakładamy w naszym programie, że start przebiegu promienia będzie po lewej stronie modelowanej soczewki. Zakładamy również pewne uproszczenie, tj. promień na początku ma kierunek zgodny z osią OX , a więc można nadać mu kierunek $\vec{k} = [1, 0]$. Przebieg zaczyna się od punktu startowego (x_0, z_0) , pada na soczewkę w punkcie (x_p, z_p) . Możemy dokładnie wyznaczyć punkt przecięcia łuku z tą prostą (promieniem świetlnym). Dalej, aby wyznaczyć kąt załamania, musimy skorzystać z wektora normalnego do łuku w naszym punkcie padania promienia na soczewkę. Wektor normalny jest liczony za pomocą następującej zależności:

$$\vec{n} = \frac{\vec{r} - \vec{r}_{rodek}}{|\vec{r} - \vec{r}_{rodek}|} = \frac{(x_p - x_c, z_p)}{R} \quad (17)$$

gdzie R – promień okręgu, na który pada promień w danej chwili
 x_c – współrzędna x-owa środka tego okręgu.

Dalej używamy odpowiednich funkcji liczących nowy kierunek wektora \vec{k} , kiedy promień przechodzi przez soczewkę oraz kiedy z niej wychodzi. Natomiast kąty załamania na kolejnym ośrodku liczymy z prawa Snella:

$$n_1 \sin(\theta_1) = n_2 \sin(\theta_2) \quad (18)$$

My natomiast w przebiegu promienia będziemy korzystać z równania parametrycznego z parametrem t (*to nie czas!*), gdzie $t \in \mathbb{R}$:

$$\vec{r}(t) = \vec{r}_0 + t\vec{k} \quad (19)$$

$$\vec{r}(t) = \begin{pmatrix} x(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} x_0 \\ z_0 \end{pmatrix} + t \begin{pmatrix} k_x \\ k_z \end{pmatrix}$$

$$\begin{cases} x(t) = x_0 + t k_x \\ z(t) = z_0 + t k_z \end{cases}$$

Podstawiając te składowe do równania okręgu otrzymujemy:

$$(x - x_c)^2 + z^2 = R^2 \quad (20)$$

$$(x_0 + tk_x - x_c)^2 + (z_0 + tk_z)^2 = R^2 \quad (21)$$

$$\text{inaczej: } at^2 + bt + c = 0 \quad (22)$$

$$\text{gdzie:} \quad (23)$$

$$a = k_x^2 + k_z^2 \quad (24)$$

$$b = 2((x_0 - x_c)k_x + z_0 k_z) \quad (25)$$

$$c = (x_0 - x_c)^2 + z_0^2 - R^2 \quad (26)$$

Z tego wynika, że mamy dwa możliwe przecięcia z okręgiem, ale tylko jedno będzie w jego widzocznej części (na soczewce) - wybieramy najmniejsze dodatnie.

W przypadku promienia padającego na płaską powierzchnię, a nie na fragment okręgu, parametr t można wyrazić dużo prościej:

$$t = \frac{(x_{\text{płaszczyzny}} - x_0)}{k_x} \quad (27)$$

gdzie $x_{\text{płaszczyzny}}$ to równanie prostej, która zamyka płaską soczewkę.

Funkcje liczące nam promień zwracają nowy punkt przecięcia oraz nowy kierunek.

Podsumowując: tak możemy wymodelować *ray tracing* dla soczewek **grubych**.

4 Opis wykorzystywanych narzędzi

- Wersja Pythona: 3.14,
- Wykorzystywane biblioteki: numpy, sympy, matplotlib, scipy,
- IDE: Visual Studio Code

5 Ogólny opis i możliwe alternatywy

Przed wyświetleniem wizualizacji użytkownik będzie miał możliwość wyboru jednej z czterech typów soczewek za pomocą przycisków. Soczewki dwuwypukłe, soczewki dwuwklesłe, soczewki płasko-wypukłe oraz płasko-wklesłe. Wtedy też program pozwoli użytkownikowi na dobranie parametru promienia światlnego - punkt jego 'startu' oraz promienie krzywizn/krzywizny za pomocą suwaku. Następnie dla wybranej soczewki program pokaże przebieg promienia światła od początku do załamania promienia z powierzchnią soczewki, przez przebieg promienia przez soczewkę, do ponownego załamania promienia na granicy soczewka-powietrze oraz dalej w nowym oknie.

Użytkownik powinien mieć możliwość wprowadzenia danych bezpośrednio w interfejsie za pomocą suwaków oraz wcześniej wspomnianych przycisków.

Jak ma wyglądać interfejs? Na początku wybranie soczewki i jej materiału, zadanie promienia krzywizny/krzywizn. Kolejne okno: wykres soczewki wraz z przebiegiem promienia w *matplotlib*

Alternatywą jest przygotowanie soczewek o odgórnie zadanych parametrach, przez którą załamywać się będzie promień. Inną alternatywą jest uruchomienie interfejsu w pojedynczym oknie, pozwolenie użytkownikowi na dobranie typów i materiałów soczewek w czasie rzeczywistym, a więc dodatnie funkcji aktualizującej obliczenia w czasie rzeczywistym. Jeszcze inną alternatywą jest dodatkowo dodanie soczewek grubych.

6 Specyficzne wymagania

6.1 Wymagania funkcjonalne

- Interfejs graficzny z obrazowym przedstawieniem soczewek,
- Wprowadzanie danych przez użytkownika,
- Wizualizacja toru promienia,

- Obliczanie załamań promieni świetlnych na soczewce,
- Obsługa błędów.

6.2 Wymagania niefunkcjonalne

- Intuicyjny interfejs graficzny,
- Modyfikowalność.

7 Harmonogram prac z zadaniami do wykonania

Tydzień	Zakres prac
1. 8 XII 2025 - 14 XII 2025	Funkcje modelujące soczewki w <i>matplotlib</i>
2. 15 XII 2025 - 4 I 2026	Dodanie funkcji załamania promienia na soczewce
3. 5 I 2026 - 11 I 2026	Interfejs graficzny, wprowadzanie danych i obsługa błędów
4. 12 I 2026 - 18 I 2026	Naprawa błędów i finalizacja

Tabela 1: Harmonogram prac

Bibliografia

- [1] *Fizyka. Krótki kurs*, Czesław Bobrowski, Wydawnictwo Naukowe PWN 2016.
- [2] *Podstawy fizyki cz. 4*, D. Halliday, R. Resnick, J. Walker, Wydawnictwo Naukowe PWN 2014.

8 Lista zmian w dokumentacji

Wersja 1.2

1. Zmiana marginesów dokumentu
2. Dodanie matematycznego opisu wymodelowanych soczewek w nowej sekcji *Modelowanie soczewek*
3. Modyfikacja tabeli z harmonogramem prac
4. Nowa bibliografia
5. Strona tytułowa

Wersja 1.3

1. Dodanie opisu przebiegu promienia świetlnego
2. Usunięcie zbędnego opisu matematycznego tworzenia obrazu przez daną soczewkę, ponieważ nie to jest modelowane